

ROBOTICS RESEARCH AND TECHNOLOGY

Parallel Manipulators

*DESIGN, APPLICATIONS AND
DYNAMIC ANALYSIS*

Cecilia Norton
Editor

NOVA

ROBOTICS RESEARCH AND TECHNOLOGY

PARALLEL MANIPULATORS

DESIGN, APPLICATIONS AND DYNAMIC ANALYSIS

No part of this digital document may be reproduced, stored in a retrieval system or transmitted in any form or by any means. The publisher has taken reasonable care in the preparation of this digital document, but makes no expressed or implied warranty of any kind and assumes no responsibility for any errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of information contained herein. This digital document is sold with the clear understanding that the publisher is not engaged in rendering legal, medical or any other professional services.

ROBOTICS RESEARCH AND TECHNOLOGY

Additional books in this series can be found on Nova's website
under the Series tab.

Additional e-books in this series can be found on Nova's website
under the e-book tab.

ROBOTICS RESEARCH AND TECHNOLOGY

PARALLEL MANIPULATORS

DESIGN, APPLICATIONS AND DYNAMIC ANALYSIS

CECILIA NORTON
EDITOR



Copyright © 2016 by Nova Science Publishers, Inc.

All rights reserved. No part of this book may be reproduced, stored in a retrieval system or transmitted in any form or by any means: electronic, electrostatic, magnetic, tape, mechanical photocopying, recording or otherwise without the written permission of the Publisher.

We have partnered with Copyright Clearance Center to make it easy for you to obtain permissions to reuse content from this publication. Simply navigate to this publication's page on Nova's website and locate the "Get Permission" button below the title description. This button is linked directly to the title's permission page on copyright.com. Alternatively, you can visit copyright.com and search by title, ISBN, or ISSN.

For further questions about using the service on copyright.com, please contact:

Copyright Clearance Center

Phone: +1-(978) 750-8400

Fax: +1-(978) 750-4470

E-mail: info@copyright.com.

NOTICE TO THE READER

The Publisher has taken reasonable care in the preparation of this book, but makes no expressed or implied warranty of any kind and assumes no responsibility for any errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of information contained in this book. The Publisher shall not be liable for any special, consequential, or exemplary damages resulting, in whole or in part, from the readers' use of, or reliance upon, this material. Any parts of this book based on government reports are so indicated and copyright is claimed for those parts to the extent applicable to compilations of such works.

Independent verification should be sought for any data, advice or recommendations contained in this book. In addition, no responsibility is assumed by the publisher for any injury and/or damage to persons or property arising from any methods, products, instructions, ideas or otherwise contained in this publication.

This publication is designed to provide accurate and authoritative information with regard to the subject matter covered herein. It is sold with the clear understanding that the Publisher is not engaged in rendering legal or any other professional services. If legal or any other expert assistance is required, the services of a competent person should be sought. FROM A DECLARATION OF PARTICIPANTS JOINTLY ADOPTED BY A COMMITTEE OF THE AMERICAN BAR ASSOCIATION AND A COMMITTEE OF PUBLISHERS.

Additional color graphics may be available in the e-book version of this book.

Library of Congress Cataloging-in-Publication Data

ISBN 978-1-63485-953-0 (eBook)

Published by Nova Science Publishers, Inc. † New York

CONTENTS

Preface		vii
Chapter 1	Adaptive Control of Parallel Manipulators: Design and Real-Time Experiments <i>M. Bennehar, A. Chemori, S. Krut and F. Pierrot</i>	1
Chapter 2	A Redundant Dynamic Modelling Procedure Based on Extra Sensors for Parallel Robot Control <i>Asier Zubizarreta, Itziar Cabanes, Marga Marcos, Charles Pinto, Javier Corral and Pablo Bengoa</i>	41
Chapter 3	Structure Synthesis of Fully-Isotropic Two-Rotational and Two-Translational Parallel Robotic Manipulators <i>Yanbin Zhang</i>	81
Chapter 4	The Parallel Two-Legged Walking Robot CENTAUROB <i>Shucen Du, Josef Schlattmann, Stefan Schulz and Arthur Seibel</i>	101
Chapter 5	Sliding-Mode Tracking Control of the 6-DOF 3-Legged Wide-Open Parallel Robot <i>Mohammad H. Abedinnasab, Jaime Gallardo-Alvarado, Bahram Tarvirdizadeh and Farzam Farahmand</i>	143
Index		167

PREFACE

Robots are a key element in current industrial processes, as they can be applied to a number of tasks, increasing both quality and productivity. Traditionally, serial robots have been installed in factories, as their wide operating space allowed them to fulfill a number of tasks. However, due to their high moving mass and single kinematic chain structure, these robots present some disadvantages when high speed, accuracy or heavy load handling tasks have to be executed. Parallel robots provide an interesting alternative to these application fields, as their multiple kinematic chain structure offers increased stiffness, allowing reduced positioning errors, lighter mechanisms and increased load/weight ratios. In this book, Chapter One addresses a new control strategy for parallel manipulators based on L_1 adaptive control. This latter is known for its decoupled control and estimation loops, enabling fast adaptation and guaranteed robustness. Chapter Two focuses on the control of parallel robots. Chapter Three reviews structure synthesis of fully-isotropic two-rotational and two-translational parallel robotic manipulators. Chapter Four reviews the new prototype of the two-legged, parallel kinematic walking robot CENTAUROB, developed at Hamburg University of Technology. Chapter Five analyzes and robustly controls the 6-DOF 3-legged Wide-Open parallel manipulator, using a Lyapunov analysis approach.

In Chapter 1, the authors address a new control strategy for parallel manipulators based on $\mathcal{L}_1$ adaptive control. This latter is known for its decoupled control and estimation loops, enabling fast adaptation and guaranteed robustness. To improve the tracking performance of parallel manipulators, the authors propose in this work to include the dynamic model of the robot in the control loop of $\mathcal{L}_1$ adaptive control. The additional

dynamics-based term partially compensates for inherent nonlinear dynamics of the robot in order to reduce the impact of uncertainties on the closed-loop system and enhance the overall control performance. Real-time experiments, conducted on a 4-DOF fast parallel manipulator developed in the laboratory, demonstrate the effectiveness of the proposed controller and its superiority compared to standard $\mathcal{L}_1$ adaptive control in terms of tracking performance.

As explained in Chapter 2, robots are a key element in current industrial processes, as they can be applied to a number of tasks, increasing both quality and productivity. Traditionally, serial robots have been installed in factories, as their wide operating space allowed them to fulfill a number of tasks. However, due to their high moving mass and single kinematic chain structure, these robots present some disadvantages when high speed, accuracy or heavy load handling tasks have to be executed.

Parallel robots provide an interesting alternative to these application fields, as their multiple kinematic chain structure offers increased stiffness, allowing reduced positioning errors, lighter mechanisms and increased load/weight ratios. These capabilities make these robots suitable for those tasks where high accelerations and speeds, micrometric precision or heavy load handling are required.

In Chapter 2, a solution based on the use of extra sensors attached to the unactuated joints of the parallel robot is detailed. The extra information provided by these sensors can be used to implement a model-based control approach that combines the performance of advanced control techniques and the robustness of sensor redundancy. In order to implement this approach, three major areas will be analyzed in the next sections: dynamic modelling, control law definition, and sensitivity, considering the well-known Gough-Stewart platform as study case.

Strong kinematic coupling is one of inherent properties for general parallel robotic manipulators. Although it contributes high stiffness and large loading capability, it also leads to complicated kinematic solutions, difficult controlling design and path planning, which will take passive effects for the actual applications of the parallel manipulators. A new systematic method for structure synthesis of four-degree-of-freedom fully-isotropic parallel robotic manipulators is proposed in Chapter 3. The moving platform of the mechanism consists of two-rotational and two-translational (2R2T) motion components with respect to the fixed base. Firstly, structure synthesis of uncoupled 2R2T parallel robotic manipulator is set up based on the concept of hybrid mechanism and the reciprocal screw theory for the first time. Then, type

synthesis of fully-isotropic 2R2T parallel robotic manipulator is performed by changing one kinematical chain's structure of the uncoupled parallel robotic manipulator obtained before. A lot of novel uncoupled and fully-isotropic 2R2T parallel robotic manipulators are synthesized. Kinematical Jacobian of a fully-isotropic 2R2T parallel robot is a 4×4 identity matrix, so there is a one-to-one linear mapping correspondence between the output velocity of the platform and the input velocity of the actuated joints. Therefore, one independent output of the moving platform can be controlled only by one actuator.

In Chapter 4, the authors review the new prototype of the two-legged, parallel kinematic walking robot CENTAUROB, developed at Hamburg University of Technology. The main features of this robot are its modular, symmetric structure, the construction of the legs as Stewart platforms (hexapods), and the special C-shaped feet that allow a statically stable movement at any time. The symmetric structure of the robot leads to a center of gravity exactly in the middle of the hip and prevents from a preferred moving direction or an asymmetric load of the legs. As a highly flexible walking device, the CENTAUROB is able to walk in unpaved environment, avoid and overcome obstacles, and even handle straight and circular stairs.

In Chapter 5, the authors analyze and robustly control the 6-DOF 3-legged Wide-Open parallel manipulator, using a Lyapunov analysis approach. The mechanism is evaluated in terms of kinematics, workspace, and dynamics. By implementing the dynamic model, tracking control of the moving platform is performed using the nonlinear sliding-mode control method. The numerical simulations with desired trajectory inputs attest to the excellent performance and robustness of the designed position tracking sliding-mode controller, in presence of fast varying uncertain parameters and external disturbances, which are common in parallel manipulators.

Chapter 1

ADAPTIVE CONTROL OF PARALLEL MANIPULATORS: DESIGN AND REAL-TIME EXPERIMENTS

M. Bennehar^{*}, *A. Chemori*[†], *S. Krut*[‡] and *F. Pierrot*[§]
LIRMM, Montpellier, France

Abstract

In this chapter, we address a new control strategy for parallel manipulators based on $\mathcal{L}_1$ adaptive control. This latter is known for its decoupled control and estimation loops, enabling fast adaptation and guaranteed robustness. To improve the tracking performance of parallel manipulators, we propose in this work to include the dynamic model of the robot in the control loop of $\mathcal{L}_1$ adaptive control. The additional dynamics-based term partially compensates for inherent nonlinear dynamics of the robot in order to reduce the impact of uncertainties on the closed-loop system and enhance the overall control performance. Real-time experiments, conducted on a 4-DOF fast parallel manipulator developed in our laboratory, demonstrate the effectiveness of the proposed controller and its superiority compared to standard $\mathcal{L}_1$ adaptive control in terms of tracking performance.

Keywords: Parallel manipulators, adaptive control, nonlinear system, dynamics

^{*}E-mail address: bennehar@lirmm.fr

[†]E-mail address: chemori@lirmm.fr

[‡]E-mail address: krut@lirmm.fr

[§]E-mail address: pierrot@lirmm.fr

1. Introduction

Adaptive control of parallel manipulators has been a decades-long area of research. It is a very relevant control strategy when it comes to control such systems due to the abundance of uncertainties and variations of their dynamics and the environment they interact with. Undoubtedly, using model-based controllers with constant dynamic parameters (e.g. computed torque [1], augmented PD [2], PD+ [3], etc.) significantly improves the overall performance of parallel manipulators and enhance their tracking capabilities. However, such control schemes are not endowed with automatic adjustment mechanisms (adaptation) to cope with variations and uncertainties in the dynamics of the controlled system and, hence, do not yield the expected performance. As a result, adaptive control seems to be a promising solution to account for such limitation by estimating and compensating the uncertainties in real-time to provide improved tracking and enhanced closed-loop performance.

The first attempts to apply adaptive control to mechanical manipulators were based on restrictive assumptions and hypotheses about their dynamics not necessarily reflecting the real properties of these systems. For instance, the proposed controller in [4], which is based on Model Reference Adaptive Control (MRAC), does not consider the coupling between the joints. Such hypothesis is questionable since mechanical manipulators are known for their coupled nonlinear dynamics. To relax this requirement, another MRAC-based controller has been proposed in [5]. The full dynamics of the manipulator as well as the coupling between the joints were explicitly considered and new adaptive laws were accordingly derived. Unfortunately for this controller, the number of parameters to be estimated is high, even for simple manipulators (11 parameters for the provided 3-DOF serial manipulator example).

One major reason of the failure of MRAC-based controllers lies in the fact that the nonlinear structure of the dynamics of the system is not taken into account. Considering this weakness, many researchers have investigated alternative adaptive control architectures inspired from non-adaptive model-based controllers. In [6], an adaptive version of the computed torque scheme has been proposed. A stability analysis based on the Lyapunov theory was conducted to derive the adaptation law that provided both the tracking of the desired trajectories as well as the convergence of the estimated parameters. Nevertheless, in this work, the control law showed two main limitations, namely the need of the measured actual joint accelerations and the boundedness of the inverse of the

estimated inertia matrix. These two limitations were bypassed in [7] by online estimating the measured accelerations while the former issue has been solved by following an approach inspired from robust control theory. Similar adaptive approaches can be found in [8, 9].

Recently, a new control scheme called $\mathcal{L}_1$ adaptive control which seeks to revisit some primary issues in MRAC has been proposed [10, 11]. This control approach, being itself inspired from MRAC, includes a low-pass filtering technique that decouples the estimation and control loops. This is of a tremendous importance since increasing the adaptation gain in conventional adaptive control may hurt the robustness of the closed-loop system. Indeed, control signals in MRAC may have undesired high frequencies lying outside the control channel bandwidth. The low-pass filter in $\mathcal{L}_1$ adaptive control, which is designed by using the $\mathcal{L}_1$ small gain theorem, is consequently introduced to remove these undesired frequencies. The first validations of $\mathcal{L}_1$ adaptive control were through numerical simulations and real-time experiments mainly in aerial vehicles [12]. It was afterwards extended to address more applications such as underwater vehicles [13], mechanical manipulators [14], underactuated systems [15] and parallel manipulators [16]. One major advantage of the $\mathcal{L}_1$ adaptive control is its being model-free; i.e. not requiring any knowledge about the dynamics of the system. However, it would be interesting to include, if available, some knowledge about the dynamics of the system in the control loop in order to enhance its tracking performance and improve the overall closed-loop system.

This chapter focuses on the development of a new control scheme based on the $\mathcal{L}_1$ adaptive control. The nominal nonlinear dynamics of the system are included in the control loop in order to improve its performance. Specifically, the proposed controller has a major advantage of improving the tracking performance which is crucial in most robotic applications. In order to demonstrate our claims, real-time experiments are conducted on Veloce; a 4-DOF parallel manipulator developed in our laboratory intended to be used for high-speed pick-and-place applications. The remainder of this chapter is organized as follows. In section 2, we provide a brief state of the art on adaptive controllers proposed in the literature for parallel manipulators. Section 3 is devoted to the modeling and description of Veloce robot. In section 4, the control problem formulation and development of $\mathcal{L}_1$ adaptive control are provided. Section 6 is dedicated to the development of the new proposed control scheme which is based on $\mathcal{L}_1$ adaptive control. Real-time experimental results are presented and discussed in section 7. Finally, conclusions are drawn in section 8.

2. Overview on Adaptive Control of Parallel Manipulators

Adaptive controllers are those control schemes requiring a real-time adjustment (i.e. adaptation) of their parameters in the aim of finding their steady-state values. The development of adaptive strategies is motivated by the abundance of uncertainties in the controlled system and its environment that may deteriorate the control performance. Adaptive controllers can be divided into two categories depending on their requirement of a dynamic model of the robot; non-model-based and model-based strategies.

In what follows, we give a brief overview on the most prominent adaptive control schemes for parallel manipulators that can be found in the literature.

2.1. Non-Model-Based Adaptive Strategies

Strategies belonging to this class do not require the dynamic model of the robot nor its structure to be known. Instead, they consider all the nonlinearities including those of the dynamic model, uncertainties and possible disturbances, whose structure is unknown, as a general disturbance term to be estimated in real-time. Then, the control law is designed such that the effect of this disturbances is minimized.

2.1.1. MRAC-Based Strategies

The main idea of MRAC is to obtain a closed-loop system with adjustable controller parameters. These parameters have the ability of changing the behavior of the closed-loop system. The time evolution of the adaptive parameters is adjusted by comparing the output of the controlled system and the desired one obtained from the reference model. The ultimate goal is to make the behavior of the controlled system match a desired reference model despite the eventual variations/uncertainties in the system or its environment [17].

While numerous MRAC-based control schemes for serial manipulators can be found in the literature [19, 20], only few controllers were proposed for parallel ones. In [18], an adaptive control scheme based on MRAC has been proposed to control a 6-DOF parallel manipulator based on the Stewart platform prototype used to emulate space operations. In this work, the control scheme consisted of a joint space PD feedback controller with adjustable gains. The control scheme

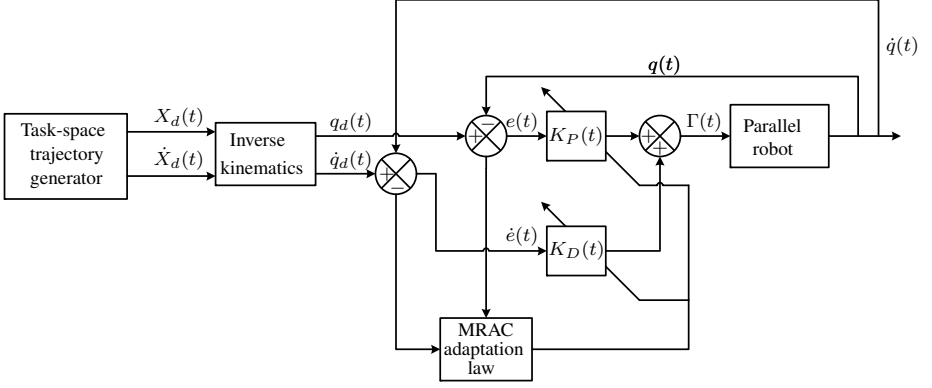


Figure 1. Bloc diagram of the proposed MRAC-based control in [18].

is further endowed with an adaptation mechanism that controls the evolution of the adaptive feedback gains. The joint tracking error of each axis was supposed to follow a desired user-defined reference second order linear system characterized by its natural frequency and its damping ratio. Based on *Lyapunov* stability analysis, the adaptation law stabilizing the error dynamics is derived. Experiments were conducted on a Stewart-platform parallel manipulator to evaluate the performance of the controller and its robustness towards sudden payload changes. Two case studies were considered; mainly, a vertical motion of the moving platform and a circular one. In both cases the proposed MRAC-based strategy with adaptive gains outperforms the PD with constant feedback gains. Figure 1 illustrates the block diagram summarizing the MRAC-based controller proposed in [18].

2.1.2. Strategies Based on Artificial Neural Networks

Artificial Neural Networks (ANN) are known by their powerful universal approximation features [21]. This is why they attracted a big deal of interest in various fields, and have been applied almost everywhere, from identification to estimation and control. Artificial neural networks learn from experience rather than programming, hence, they are typically used in repetitive tasks. Several works can be found in the literature regarding the application of neural networks in control of parallel manipulators [22]. However they remain relatively few compared to what can be found for serial manipulators [23].

Thanks to their learning ability, artificial neural networks are mostly used to approximate the dynamics of the manipulator. Then, the learned dynamics can be included in the control scheme to compensate for the uncertainties and disturbances. [24] proposed to augment a decentralized Cartesian space PID controller with an artificial neural network term. The main motivation of this work is to improve the tracking capabilities of a 2-DOF redundantly actuated parallel manipulator. The provided simulation results demonstrated the superiority of the augmented controller with respect to the original PID. The maximum errors were significantly reduced since the additional neural network term accounted for the nonlinear dynamics of the manipulator.

2.2. Model-Based Adaptive Strategies

Model-based adaptive controllers explicitly include the dynamics of the robot in the control loop. In contrast to their non-adaptive counterparts, the adaptive schemes adjust the parameters of the model-based loop in order to converge them their best steady-space values. Consequently, if the dynamics of the manipulator is uncertain or time-varying, adaptive controllers result in a better closed-loop performance.

2.2.1. Strategies Based on Computed Torque

Computed-torque-based adaptive schemes are those controllers relying on the classical non-adaptive computed torque controller. The control design starts by considering the ideal non-adaptive controller that assumes perfect cancellation of the system nonlinearities. Due to possible uncertainties and variations in the system dynamics, the compensation of the nonlinearities of the system could not be as perfect as the non-adaptive controller assumes. Then, an additional adaptive estimation loop is considered to account for these uncertainties. The role of the adaptive loop is to estimate, in real-time, the system's parameters and the obtained estimation are used in the controller.

The typical example of a computed-torque-based adaptive controller is the one proposed in [25]. The first step of the control design in this work is to consider a computed torque control law with uncertain parameters. Then, the error equation resulting from the application of this control law is obtained. Based on a thorough stability analysis using the *Lyapunov* theory, an adaptive law for the parameters' estimation is derived. The proposed adaptive controller in conjunction with the adaptation law guarantee that the tracking errors vanish

and that the estimated parameters converge to their best steady-state values. The block diagram depicted in Figure 2 clarifies the principle of such control strategy.

The computed-torque-based adaptive controller, proposed in [25], was mainly developed for serial manipulators. However, it is known that parallel manipulators share many properties with their serial counterparts [26, 27]. Based on this fact, the controller in [25] has been straightforwardly applied to PKMs in [28]. In this work, an adaptive Cartesian space computed-torque-based scheme is applied to a 2-DOF redundantly actuated parallel manipulator. Real-time experiments were carried out in order to highlight the benefits of the adaptive controller compared to its non-adaptive version. The adaptive controller shows a net superiority and allows to estimate the system dynamic parameters.

A main drawback of computed torque based adaptive controllers is their dependence on the real acceleration of the robot [25]. This shortcoming is crucial from a practical point of view since measuring actual accelerations is tedious.

2.2.2. Based on Passivity of the System

Passivity-based controllers use the passivity property of the manipulator [7]. In contrast with computed-torque-based ones, passivity-based controllers do not assume perfect cancellation of the system nonlinearities, even in the ideal case of perfect knowledge of the manipulator's parameters. Hence, they do not lead to a closed-loop linear error system. However, based on the passivity property of the system, they result in a stable closed-loop system.

[8] proposed a passivity-based adaptive controller that holds many advantageous features. This controller can be assimilated to an adaptive version of PD

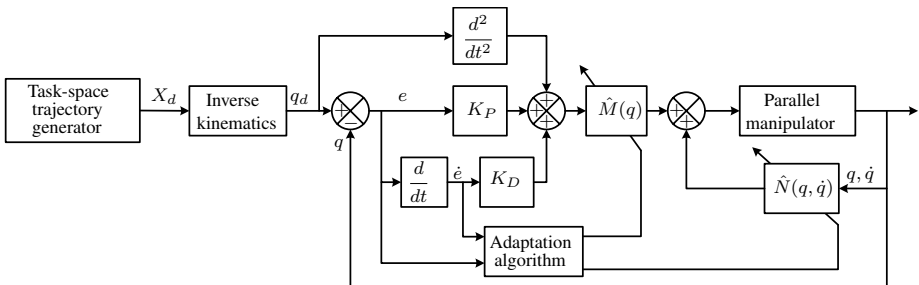


Figure 2. Bloc diagram of adaptive computed torque control.

control with computed feedforward. Indeed, the proposed control law consists mainly in a PD feedback loop in addition to an adaptive feedforward term based on the dynamics of the robot and the desired reference trajectories. The main advantage of this scheme is that it does not require the joint acceleration measurement. Moreover, the compensation of the system's nonlinearities is based on desired quantities that can be computed offline. This means that the control scheme does not require heavy computations. Furthermore, the use of the desired values instead of the measured ones, which can be often noisy, may enhance the robustness of the controller toward measurement noise.

In [29], the developed controller in [8] has been embraced for joint space control of a 6-DOF parallel manipulator called the Hexaglide. The proposed adaptive controller as well as a linear PD controller were implemented in real-time for a comparison purpose. It was not surprising that the adaptive controller provided better tracking results since it compensates for the nonlinear dynamics of the manipulator.

In [30], a Cartesian space control strategy called *Dual-Space Adaptive Control* is proposed to control redundantly actuated parallel manipulators. The proposed controller in this work share many similarities with the proposed control strategy in [29]. The main difference is that the weighting gain of the position error with respect to the velocity error is time-varying. The proposed controller was experimentally validated on the R4 parallel manipulator [31]. Real-time experiments showed that the proposed Dual-Space adaptive controller significantly improves the tracking performance and allows the tracking of extremely fast trajectories (up to 100G of maximum acceleration).

Sadegh [32] proposed to extend the joint-space passivity based adaptive controller of [33], mainly developed for serial manipulators, to Cartesian-space control of parallel manipulators with actuation redundancy. Moreover, to deal with friction an additional control term has been added to the conventional feedback and adaptive loops. Real-time experiments, conducted on a 2-DOF redundantly actuated parallel manipulator, showed the superiority of the proposed adaptive controller compared to the augmented PD controller in terms of Cartesian tracking errors, at both low and high accelerations. Unfortunately, the evolution of the estimated parameters was not shown in this work to see if they really converge to their steady-state values.

3. Modeling of Delta-Like Parallel Robots: Application to Veloce

Veloce is a 4-DOF Delta-like [34] fully actuated parallel manipulator having four identical kinematic chains [35]. Each kinematic chain can be seen as serial arrangement of the actuator, a rear-arm and a forearm. Its articulated moving platform illustrated in Figure 3, which is connected to the fixed-base through the kinematic chains, can perform three spatial translations and one rotation around the vertical axis. The rotation is obtained via the relative motion of the upper and lower parts of the moving platform. The four actuators responsible for the movement of the mechanical structure are all lying on the same plane. The CAD of Veloce is illustrated in Figure 4.

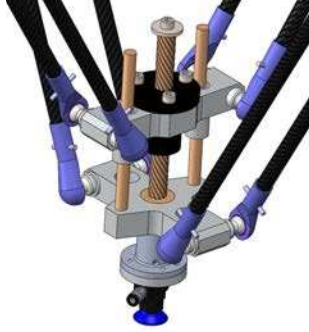


Figure 3. Articulated moving platform of Veloce.

3.1. Inverse Kinematics Model

The inverse kinematics problem consists in finding the actuated joint vector $q = [q_1, q_2, q_3, q_4]^T \in \mathbb{R}^4$ corresponding to a specific pose $X = [x, y, z, s]^T$ of the moving platform of Veloce. Figures 5 and 6 in addition to Table 1 summarize the various geometric parameters involved in the establishment of the inverse kinematic model of Veloce. The actuated joints locations are represented by points A_i , $i = 1, \dots, 4$, whose coordinates are given in the fixed Cartesian frame $O - xyz$ by:

$$A_i = r_b [\cos(\alpha_i), \sin(\alpha_i), 0]^T, \quad (1)$$

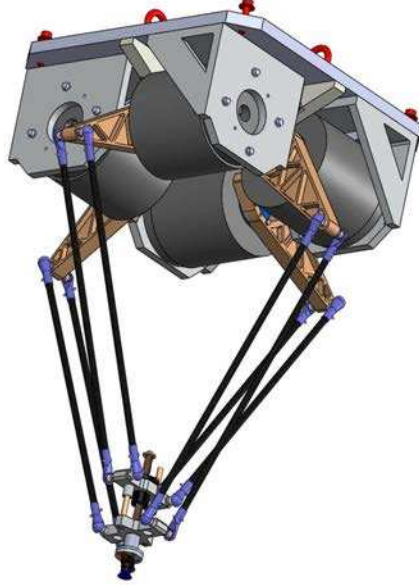


Figure 4. CAD view of Veloce 4-DOF parallel manipulator.

where $\alpha_i = (i - 1)\pi/2$, $i = 1, \dots, 4$, is the orientation of the i th actuator axis with respect to the fixed Cartesian frame. The locations of the centers of passive ball joints are represented by points B_i and C_i which are expressed in the base frame $O - xyz$ by:

$$B_i = A_i + L [\cos(\alpha_i) \cos(q_i), \sin(\alpha_i) \cos(q_i), \sin(q_i)], \quad (2)$$

$$C_i = [r_p \cos(\alpha_i) + x, r_p \sin(\alpha_i) + y, z]. \quad (3)$$

Table 1. Geometric parameters of Veloce robot

Parameter	Description	Value
L_i	Rear-arm length	200 mm
l_i	Forearm length	530 mm
r_b	Base radius	135 mm
r_p	Moving Platform radius	48 mm

To facilitate the subsequent mathematical development, a frame $P_i - u_i v_i z$ is attached to each actuator, where the vectors u_i and v_i are given by:

$$u_i = [\cos(\alpha_i), \sin(\alpha_i), 0]^T, \quad (4)$$

$$v_i = [-\sin(\alpha_i), \cos(\alpha_i), 0]^T, \quad (5)$$

where v_i is always directed towards the i th actuator's axis. The inverse kinematics solution of Veloce robot can be obtained by finding the intersection of a circle and a sphere which represents the coordinates of the passive joints $B_i = [x_{B_i}, 0, z_{B_i}]^T$ in the corresponding $A_i - u_i v_i z$ frame [36, 37]. Notice that in this frame, the B_i points always satisfy $y_{B_i} = 0$.

From the motion of one rear-arm, we can write the equation of a circle in the corresponding $A_i - u_i z$ plane of center $A_i = [0, 0]$ and of radius L as follows:

$$x_{B_i}^2 + z_{B_i}^2 = L^2 \quad (6)$$

The movements of one forearm can be described by a sphere of center C_i

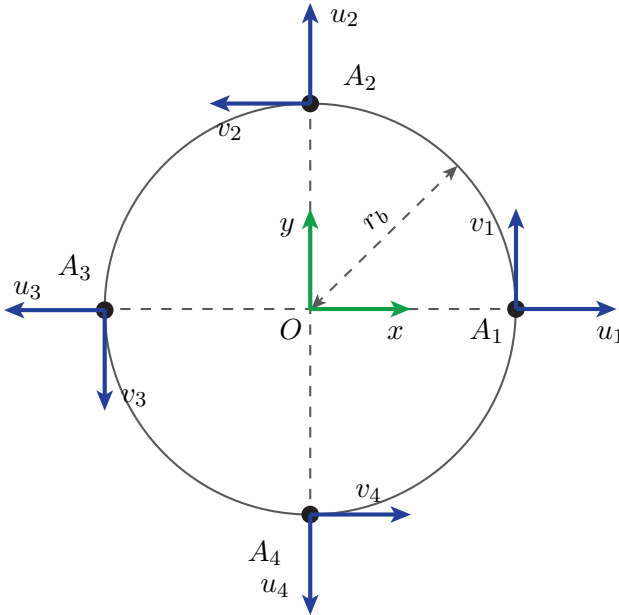


Figure 5. Geometric parameters of Veloce robot: top view.

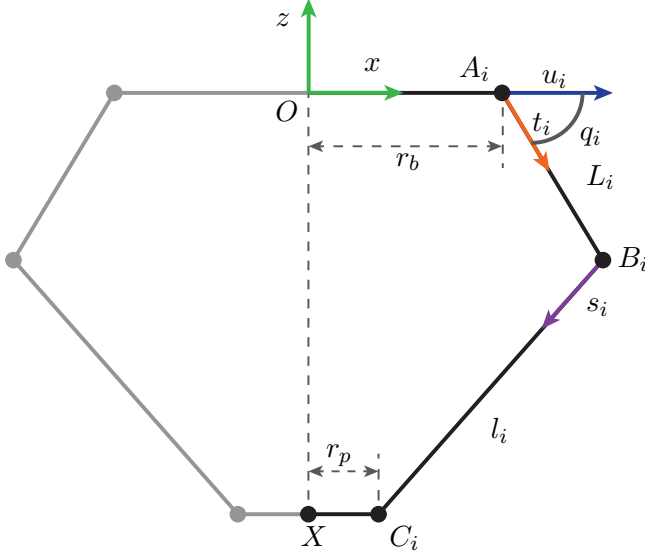


Figure 6. Geometric parameters of Veloce robot: side view.

and radius l . In the $A_i - u_i v_i z$ frame, we can write the following equation:

$$(x_{B_i} - x_{C_i})^2 + y_{C_i}^2 + (z_{B_i} - z_{C_i})^2 = l^2 \quad (7)$$

Subtracting (6) from (7) yields:

$$2x_{C_i}x_{B_i} + 2z_{C_i}z_{B_i} = l^2 - L^2 + x_{C_i}^2 + y_{C_i}^2 + z_{C_i}^2 \quad (8)$$

Solving (8) for z_{B_i} yields:

$$z_{B_i} = \frac{S_i - 2x_{C_i}}{2z_{C_i}}, \quad (9)$$

with $S_i \triangleq l^2 - L^2 + x_{C_i}^2 + y_{C_i}^2 + z_{C_i}^2$. Substituting (9) in (6) yields:

$$4(z_{C_i}^2 + x_{C_i}^2)x_{B_i}^2 - 4S_ix_{C_i}x_{B_i} + (S_i^2 - 4z_{C_i}^2L^2) = 0 \quad (10)$$

Solving for x_{B_i} results in:

$$x_{B_i} = \frac{S_ix_{C_i} \pm \sqrt{(S_ix_{C_i})^2 - (z_{C_i}^2 + x_{C_i}^2)(S_i^2 - 4z_{C_i}^2L^2)}}{2(z_{C_i}^2 + x_{C_i}^2)} \quad (11)$$

Equations (11) and (9) result in the coordinates of the intersection of a circle and a sphere. For geometrical reasons, only the largest x_{B_i} is kept. Once the coordinates of the points B_i are found, the corresponding actuated joint value can be obtained by:

$$q_i = \text{atan2}(z_{B_i}, x_{B_i}) \quad (12)$$

Applying (12) to each of the four kinematic chains results in the solution of the inverse kinematic model of Veloce.

3.2. Jacobian Matrix

The Jacobian matrix $J(q, X) \in \mathbb{R}^{4 \times 4}$ of Veloce relates its moving platform's velocity vector $\dot{X} = [\dot{x}, \dot{y}, \dot{z}, \dot{s}]$ to the actuated joints velocity vector $\dot{q} = [\dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4]$ such that:

$$\dot{X} = J\dot{q} \quad (13)$$

Note that q and X are related through the inverse kinematic model. The Jacobian matrix J can be obtained based on the following kinematic relationship:

$$\|B_i C_i\|^2 - l^2 = 0, \quad (14)$$

which means that the length of each forearm remains constant independently of the robot's configuration. Differentiating (14) and rearranging the terms results in:

$$J_x \dot{X} = J_q \dot{q}, \quad (15)$$

where J_q and J_x are given as follows:

$$J_q = \text{diag}(l_1 w_1^T s_1, \dots, l_n w_n^T s_n), \quad (16)$$

$$J_x = [s_1 \dots s_n]^T, \quad (17)$$

where

$$w_i = t_i \times v_i \quad (18)$$

$$t_i = \frac{A_i B_i}{L_i} \quad (19)$$

$$s_i = \frac{B_i C_i}{l_i} \quad (20)$$

Finally, the Jacobian matrix can be written as follows:

$$J = \begin{bmatrix} \frac{l_1 w_1^T s_1}{s_1} & \dots & \frac{l_i w_i^T s_i}{s_i} \end{bmatrix} \quad (21)$$

3.3. Inverse Dynamic Model

In this section, a brief description of the simplified dynamic model of Veloce is presented in the sequel. But first, to simplify the motion equations of the different parts of the mechanical structure of Veloce, the following assumptions, commonly used in Delta-like robots, are considered [37]

Assumption 1. *Both dry and viscous frictions in all passive and active joints are neglected. This is mainly due to the fact that the joints are carefully designed such that friction effects are minimized.*

Assumption 2. *The rotational inertia of the forearms is neglected and their mass is split up into two equivalent parts, one is added to the mass of the arm while the other one is considered with the moving platform. This hypothesis is justified by the small mass of the forearms compared to other components.*

The dynamics of the Veloce robot shows a lot of similarities with those of the Delta robot. Nevertheless few differences arise due to the number of kinematic chains and the additional rotational DOF of the moving platform.

Regarding the moving platform, we distinguish two kinds of forces acting on it, the gravity forces $G_p \in \mathbb{R}^4$ and the inertial force $F_p \in \mathbb{R}^4$, they are given by

$$G_p = M_p [0 \ 0 \ -g \ 0]^T \quad (22)$$

$$F_p = M_p \ddot{X} \quad (23)$$

Table 2. Dynamic parameters of Veloce robot

Parameter	Description	Value
m_a	One Rear-arm's weight	0.541 kg
m_f	One Forearm's weight	0.08 kg
m_p	Platform's weight	0.999 kg

where $M_p \in \mathbb{R}^{4 \times 4}$ is the mass matrix of the moving platform that also considers the half-masses of the forearms, $g = 9.81 \text{ m/s}^2$ is the gravity constant and $\ddot{X} \in \mathbb{R}^4$ is the Cartesian acceleration vector.

The contributions of G_p and F_p to each motor can be computed using the Jacobian matrix $J(q, X) \in \mathbb{R}^{4 \times 4}$ as follows

$$\Gamma_{G_p} = J^T M_p [0 \ 0 \ -g \ 0]^T \quad (24)$$

$$\Gamma_p = J^T M_p \ddot{X} \quad (25)$$

From the joints side, the elements that contribute to the dynamics of the actuators are the forces and torques resulting from the movement of the rear-arms in addition to the half-masses of the forearms.

Applying the virtual work principle, which states that the sum of non-inertial forces is equal to that of the inertial ones, and after rearranging the terms, we get

$$J^T M_p \ddot{X} + \Gamma_{G_p} + I_a \ddot{q} + \Gamma_{G_a} = \Gamma \quad (26)$$

being $I_a \in \mathbb{R}^{4 \times 4}$ a diagonal inertia matrix of the arms accounting for the rear-arms as well as the half-masses of the forearms, $\ddot{q} \in \mathbb{R}^4$ the joint acceleration vector and $\Gamma_{G_a} \in \mathbb{R}^4$ is the force vector resulting from gravity acting on the arms being given by

$$\Gamma_{G_a} = m_a r_{G_a} g \cos(q) \quad (27)$$

with m_a the sum of the mass of one rear-arm and one half-mass of a forearm, r_{G_a} the distance between the center of one axis and the center of mass of one arm and q_1, q_2, q_3 and q_4 are the joints positions.

Given the kinematic relationship $\ddot{X} = J\ddot{q} + \dot{J}\dot{q}$, (26) can be rewritten as follows

$$(I_a + J^T M_p J) \ddot{q} + \left(J^T M_p \dot{J} \right) \dot{q} - (J^T G_p + \Gamma_{G_a}) + \Gamma_d = \Gamma, \quad (28)$$

where Γ_d is a disturbance term that accounts for non-modeled dynamics and possible external disturbances. It can be written in a standard joint-space form as follows

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \Gamma_d = \Gamma, \quad (29)$$

with:

$M(q) = I_a + J^T M_p J$ the total mass matrix.

$C(q, \dot{q}) = J^T M_p \dot{J}$ the Coriolis and centrifugal forces matrix.

$G(q) = -(J^T G_p + \Gamma_{G_a})$ the gravitational forces vector.

The expression of the dynamics in (29) is suitable for joint-space control since it is expressed in terms of the actuated joints.

4. Background on $\mathcal{L}_1$ Adaptive Control

In this section, the basic idea behind the development of the $\mathcal{L}_1$ adaptive control theory is detailed. For the sake of simplicity, we consider a single-input single-output system. Since $\mathcal{L}_1$ adaptive control is inspired from MRAC, we first introduce MRAC for the considered system. Then the architecture of MRAC is changed to achieve the decoupling of robustness and adaptation.

4.1. Control Problem Formulation

Consider the following single-input single-output linear time-invariant system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + bu(t), \quad x(0) = x_0 \\ y(t) &= c^T x(t) \end{aligned} \tag{30}$$

where

$x(t) \in \mathbb{R}^n$ is the measurable state of the system,

$u(t) \in \mathbb{R}$ is the control input,

$b, c \in \mathbb{R}^n$ are known constant input and output vectors,

$y(t) \in \mathbb{R}$ is the output to be regulated by the controller.

The system state matrix $A \in \mathbb{R}^{n \times n}$ is supposed unknown. The control objective is to design an adaptive control signal $u(t)$ such that the regulated output of the system $y(t)$ tracks a given reference signal $r(t) \in \mathbb{R}$ with desired performance while guaranteeing that the system state and other signals remain bounded under the following assumption

Assumption 3. *There exist a Hurwitz matrix $A_m \in \mathbb{R}^{n \times n}$ and a vector $\theta \in \mathbb{R}^n$ of perfect parameters such that the pair (A_m, b) is controllable and $A_m - A = b\theta^T$. Moreover, assume that this unknown perfect parameters vector θ belongs to a given compact set Θ .*

Under the above assumption, the system in (30) can be rewritten as follows

$$\begin{aligned} \dot{x}(t) &= A_m x(t) + b(u(t) - \theta^T x(t)), \quad x(0) = x_0 \\ y(t) &= c^T x(t) \end{aligned} \quad (31)$$

4.2. From Direct MRAC to Direct MRAC with a State Predictor

One of the key differences between MRAC and $\mathcal{L}_1$ adaptive control is the introduction of a prediction-based adaptive architecture. In the following, this new structure is highlighted and demonstrated that it leads to the same closed-loop behavior as direct MRAC.

4.2.1. Direct MRAC

Consider the nominal nonadaptive controller given by

$$u_{nom}(t) = \theta^T x(t) + k_g r(t) \quad (32)$$

where $k_g \triangleq 1/(c^T A_m^{-1} b)$ is a feedforward gain that ensures zero steady-state tracking error. Substituting (32) in the system dynamics in (31) leads to the following desired reference system

$$\begin{aligned} \dot{x}_m(t) &= A_m x_m(t) + b k_g r(t), \quad x(0) = x_0 \\ y_m(t) &= c^T x_m(t) \end{aligned} \quad (33)$$

where $x_m(t)$ is the state of the reference system and $y_m(t)$ its output. The nominal control law is impossible to implement due to the uncertainties in the system state matrix A (and, hence, θ) which should be estimated online.

The feasible direct MRAC control law is hence given by

$$u(t) = \hat{\theta}^T x(t) + k_g r(t) \quad (34)$$

where $\hat{\theta} \in \mathbb{R}^n$ is an estimate of θ whose evolution is governed by the following adaptation rule

$$\dot{\hat{\theta}}(t) = \gamma x(t) e^T(t) P b, \quad \hat{\theta}(0) = k_{x0} \quad (35)$$

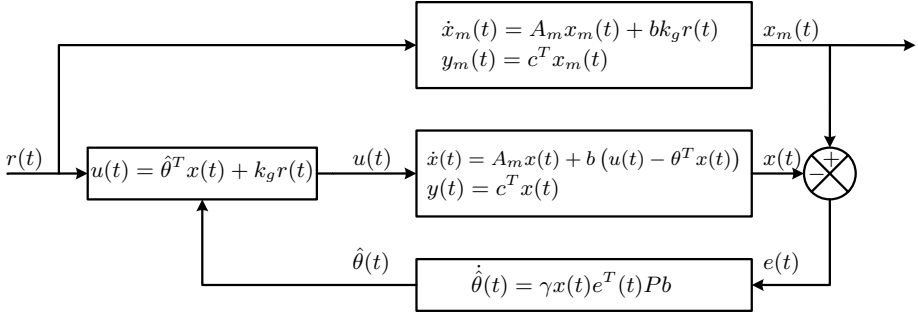


Figure 7. Block diagram of MRAC.

where $\gamma \in \mathbb{R}^+$ is the adaptation gain, $e(t) \triangleq x_m(t) - x(t)$ and $P = P^T > 0$ is the solution for the algebraic Lyapunov equation $A_m^T P + P A_m = -Q$, for an arbitrary choice of $Q = Q^T > 0$.

Substituting the adaptive control law (34) in the system dynamics in (31) leads to the following error dynamics

$$\dot{e}(t) = A_m e(t) - b \tilde{\theta}^T(t) x(t) \quad (36)$$

where $\tilde{\theta} \triangleq \hat{\theta}(t) - \theta$ is the parameters estimation error. Figure 7 depicts the overall block diagram of MRAC.

4.2.2. MRAC with a State Predictor

Now consider the following state predictor that mimics the behavior of the system in (31) with the unknown parameter θ replaced by its estimate $\hat{\theta}(t)$

$$\begin{aligned} \dot{\hat{x}}(t) &= A_m \hat{x}(t) + b(u(t) - \hat{\theta}^T x(t)), \quad \hat{x}(0) = x_0 \\ \hat{y}(t) &= c^T \hat{x}(t) \end{aligned} \quad (37)$$

where $\hat{x}(t)$ being the state of the predictor. The prediction error dynamics can be obtained by subtracting the system dynamics from those of the predictor as follows

$$\dot{\tilde{x}}(t) = A_m \tilde{x}(t) - b \tilde{\theta}^T(t) x(t) \quad (38)$$

where $\tilde{x}(t) \triangleq \hat{x}(t) - x(t)$ is the prediction error. It can be seen that the prediction error dynamics is identical to the obtained error dynamics with direct MRAC in

(36). The adaptive law of the predictor-based controller is similar to that of direct MRAC, the only difference is the use of the prediction error $\tilde{x}(t)$ instead of the tracking error $e(t)$. It is thus given by

$$\dot{\hat{\theta}}(t) = \gamma x(t) \tilde{x}^T(t) P b, \quad \hat{\theta}(0) = \theta_0 \quad (39)$$

It can be noticed that the closed-loop state predictor mimics the behavior of the reference model in (33). Indeed, substituting the control law (34) in the predictor dynamics and upon the use of (39) we get the following closed-loop dynamics

$$\begin{aligned} \dot{\hat{x}}(t) &= A_m \hat{x}(t) + b k_g r(t), \quad \hat{x}(0) = x_0, \\ \hat{y}(t) &= c^T \hat{x}(t) \end{aligned} \quad (40)$$

which is identical to the dynamics of the reference system of direct MRAC in (33). It is demonstrated in [38] that the tracking (or prediction) errors are upper bounded at any time by

$$\|e(t)\| (= \|\tilde{x}(t)\|) \leq \frac{\|\tilde{\theta}(0)\|}{\sqrt{\lambda_{\min}(P)}\gamma} \quad (41)$$

being $\tilde{\theta}(0)$ the initial parameters estimation error and $\lambda_{\min}(P)$ the minimum eigenvalue of P . This means that the tracking error can be made arbitrarily small by increasing the adaptation gain γ . However, it can be seen from the control law in (34) and the adaptation laws in (35) and (39) that increasing the adaptation gain lead to high gain feedback. The block diagram of MRAC scheme with a state predictor is shown in Figure 8.

4.3. $\mathcal{L}_1$ Adaptive Control

In a real system, the unknown parameters vector θ may be a time-varying uncertainty with frequencies that may lie outside the control channel bandwidth [38]. In this case, such frequencies are naturally attenuated due to hardware limitations of the actuators. Consequently, the behavior of the closed-loop system will be different than the dynamics of the reference model in (33).

The control problem in $\mathcal{L}_1$ adaptive control is formulated with the understanding that some uncertainties could never be perfectly compensated. Indeed, while the control objective in MRAC is only stated asymptotically, the closed-loop performance in $\mathcal{L}_1$ adaptive control is specified $\forall t \geq 0$. This means that at

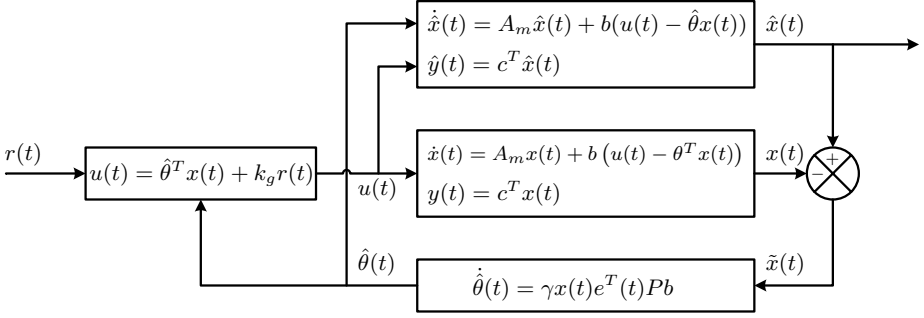


Figure 8. Block diagram of the control loop based on MRAC with state predictor.

any time, the performance of the system can be predicted. Moreover, the control signal never exceeds the available control channel bandwidth.

For the class of systems given by (31), consider the following state predictor that mimics its behavior

$$\begin{aligned}\dot{\hat{x}}(t) &= A_m \hat{x}(t) + b(u(t) - \hat{\theta}^T x(t)), \quad \hat{x}(0) = x_0 \\ \hat{y}(t) &= c^T \hat{x}(t)\end{aligned}\quad (42)$$

In addition to the state predictor in (42), consider a projection-type adaption law for the estimated parameter vector $\hat{\theta}$ expressed as

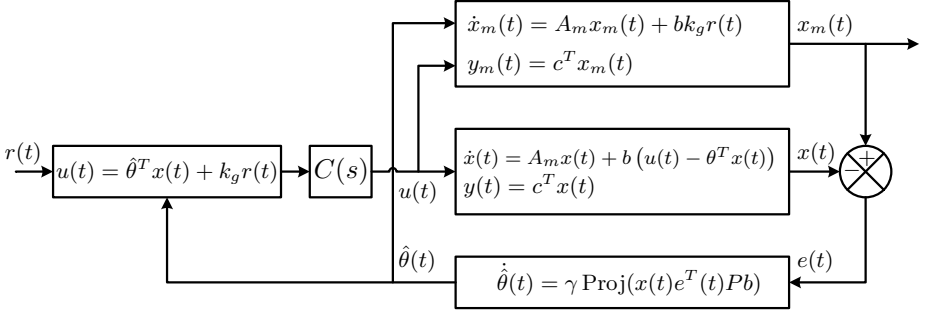
$$\dot{\hat{\theta}}(t) = \gamma \text{Proj} \left(\hat{\theta}(t), x(t) \tilde{x}(t) P b \right), \quad \hat{\theta}(0) = \theta_0 \quad (43)$$

which is adjusted using the prediction error $\tilde{x}(t) = \hat{x}(t) - x(t)$. The projection operator Proj avoids the parameters drift and ensures that they remain inside the compact set Θ . In (43), γ is the positive adaptation gain and P is the symmetric positive definite matrix, solution of the algebraic *Lyapunov* equation $A_m^T P + P A_m = -Q$ for an arbitrary symmetric positive definite matrix Q .

The last stage which is one of the notable unique features of the $\mathcal{L}_1$ adaptive control, is the control input characterized by the introduction of a low-pass filter. It is given by its *Laplace* form as follows

$$u(s) = C(s) (\hat{\eta}(s) + k_g r(s)) \quad (44)$$

where $C(s)$ is a bounded-input bounded-output strictly proper transfer function with $C(0) = 1$, $r(s)$ is the *Laplace* transform of the reference trajectory $r(t)$ and $\hat{\eta}(s)$ is the *Laplace* transform of $\hat{\eta}(t) = \hat{\theta}^T(t)x(t)$.

Figure 9. Block diagram of $\mathcal{L}_1$ adaptive control.

It is demonstrated in [38] that the controlled system in (30) under the $\mathcal{L}_1$ adaptive control law in (44) is bounded-input bounded-state with respect to $r(t)$ and x_0 if the following $\mathcal{L}_1$ -norm-based equality is satisfied

$$\|G(s)\|_{\mathcal{L}_1} L < 1 \quad (45)$$

where

$$G(s) \triangleq H(s)(1 - C(s)), \quad H(s) \triangleq (s\mathbb{I} - A_m)^{-1}b, \quad L \triangleq \max_{\theta \in \Theta} \|\theta\|_1 \quad (46)$$

The block diagram of the $\mathcal{L}_1$ adaptive control strategy for the class of systems given by (31) is displayed in Figure 9.

5. Application of $\mathcal{L}_1$ Adaptive Control to Parallel Manipulators

Recall the inverse dynamics for parallel mechanical manipulators which are expressed as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \Gamma_d(t) = \Gamma(t) \quad (47)$$

In the aim of developing a $\mathcal{L}_1$ adaptive controller for the class of systems given by (47), introduce the following combined position-velocity tracking error given by

$$r = (\dot{q} - \dot{q}_d) + \Lambda(q - q_d) \quad (48)$$

where $\Lambda \in \mathbb{R}^{4 \times 4}$ is a symmetric positive-definite weighting matrix.

5.1. Control Law

Consider the following control input vector $\Gamma(t)$ consisting of a combination of two distinct terms [39]

$$\Gamma(t) = \Gamma_m(t) + \Gamma_{ad}(t), \quad \Gamma_m(t) \triangleq A_m r(t) \quad (49)$$

where $\Gamma_m \in \mathbb{R}^n$ is a state-feedback term characterizing the desired transient response of the tracking error $r(t)$ and Γ_{ad} is the adaptive control term which will be designed subsequently. Taking the first time derivative of (48) with respect to time we get

$$\dot{r}(t) = (\ddot{q} - \ddot{q}_d) + \Lambda (\dot{q} - \dot{q}_d) \quad (50)$$

Substituting the control law (49) into the dynamics of the parallel manipulator in (47) and solving for $\ddot{q}$ along with substituting into (50), we get the following error dynamics

$$\dot{r}(t) = A_m r(t) + \Gamma_{ad}(t) - \eta(t, \zeta(t)), \quad r(0) = r_0 \quad (51)$$

where $\zeta = [r, q]^T$ and $\eta(t, \zeta(t))$ is a nonlinear function that gathers all the nonlinearities of the system including possible external disturbances. Before proceeding to the development of the adaptive control law, some important assumptions and properties regarding the unknown function $\eta(t, \zeta(t))$ have to be made, they can be summarized as follows

Assumption 4. (Uniform boundedness of $\eta(t, 0)$) There exist $B > 0$ such that $\forall t \geq 0, \|\eta(t, 0)\| \leq B$.

Assumption 5. (Semiglobal uniform boundedness of partial derivatives of $\eta(t, \zeta(t))$) The unknown nonlinear function $\eta(t, \zeta(t))$ is continuous with respect to its arguments, and for arbitrary $\delta > 0$, there exist $d_{\eta_t}(\delta)$ and $d_{\eta_x}(\delta)$ such that

$$\left\| \frac{\partial \eta(t, \zeta)}{\partial t} \right\|_{\infty} \leq d_{\eta_t}(\delta), \quad \left\| \frac{\partial \eta(t, \zeta)}{\partial \zeta} \right\|_{\infty} \leq d_{\eta_x}(\delta) \quad (52)$$

Assumption 6. for $t \geq 0$, $\|r_t\|_{\mathcal{L}_{\infty}} \leq \rho$ and $\|\dot{r}_T\|_{\mathcal{L}_{\infty}} \leq d_r$, for some positive constants ρ and d_r .

It follows from Lemma A.9.2 in [38] that the unknown nonlinear function $\eta(t, \zeta(t))$ can be parametrized as follows

$$\eta(t, \zeta(t)) = \theta(t) \|r_t\|_{\mathcal{L}_\infty} + \sigma(t) \quad (53)$$

where $\theta(t), \sigma(t) \in \mathbb{R}^n$ are continuous, piecewise-differentiable and uniformly bounded unknown functions. Therefore, the error dynamics in (51) can be rewritten as:

$$\dot{r}(t) = A_m r(t) + \Gamma_{ad}(t) - (\theta(t) \|r_t\|_{\mathcal{L}_\infty} + \sigma(t)), \quad r(0) = r_0. \quad (54)$$

Notice that the substitution of $\Gamma_{ad}(t) = (\theta(t) \|r_t\|_{\mathcal{L}_\infty} + \sigma(t))$ in the error dynamics in (54) results in the desired performance characterized by $\dot{r}(t) = A_m r(t)$. However, since the nonlinear function $\eta(t, \zeta(t)) = (\theta(t) \|r_t\|_{\mathcal{L}_\infty} + \sigma(t))$ is unknown due to uncertainties and unmeasured disturbances, the control term $\Gamma_{ad}(t)$ should be designed in a way that it estimates the unknown functions $\theta(t)$ and $\sigma(t)$ in real-time.

5.2. State Predictor

To predict the behavior of the combined tracking error $r(t)$ and according to the $\mathcal{L}_1$ adaptive control theory, a state predictor that mimics the behavior of the error dynamics in (54) is formulated by [39]:

$$\dot{\hat{r}}(t) = A_m \hat{r}(t) + \Gamma_{ad}(t) - \left(\hat{\theta}(t) \|r_t\|_{\mathcal{L}_\infty} + \hat{\sigma}(t) \right) - Z \tilde{r}(t), \quad \hat{r}(0) = r_0 \quad (55)$$

where $\hat{r}(t)$ is the state of the predictor, $\tilde{r}(t) \triangleq \hat{r}(t) - r(t)$ is the prediction error and $\hat{\theta}(t), \hat{\sigma}(t)$ are estimates of $\theta(t)$ and $\sigma(t)$, respectively. Notice the introduction of the additional term $Z \tilde{r}(t)$ responsible of rejecting the estimation error.

5.3. Adaptation Laws

The evolution of the adaptive estimates $\hat{\theta}(t), \hat{\sigma}(t)$, required for the $\mathcal{L}_1$ adaptive control architecture, is governed by the following projection-type adaptation laws

$$\dot{\hat{\theta}}(t) = \Xi \text{Proj} \left(\hat{\theta}(t), P \tilde{r}(t) \|r_t\|_{\mathcal{L}_\infty} \right), \quad \hat{\theta}(0) = \hat{\theta}_0 \quad (56)$$

$$\dot{\hat{\sigma}}(t) = \Xi \text{Proj} \left(\hat{\sigma}(t), P \tilde{r}(t) \right), \quad \hat{\sigma}(0) = \hat{\sigma}_0 \quad (57)$$

the combined joint tracking error $r(t) \in \mathbb{R}^4$ defined by

$$r(t) = (\dot{q} - \dot{q}_d) + \Lambda (q - q_d) \quad (59)$$

where $\Lambda \in \mathbb{R}^{4 \times 4}$ is a symmetric positive-definite design matrix. Since $-\Lambda$ is Hurwitz, it follows that (59) is BIBO-stable; i.e. there exist $L_1, L_2 > 0$ such that [40]

$$\|q_\tau\|_{\mathcal{L}_\infty} \leq L_1 \|r_\tau\|_{\mathcal{L}_\infty} + L_2 \quad (60)$$

where $\|(\cdot)_\tau\|_{\mathcal{L}_\infty}$ denotes the truncated $\mathcal{L}_\infty$ -norm of $(\cdot)$. The proposed control law to achieve the desired tracking performance is given by:

$$\Gamma = M_0(q_d)\ddot{q}_d + C_0(q_d, \dot{q}_d)\dot{q}_d + G_0(q_d) + A_m r + \Gamma_{ad}, \quad (61)$$

The proposed control law in (61) can be thought of as the combination of three distinct terms. Each term has a specific role in the closed-loop system:

- The first term (i.e. $M_0(q_d)\ddot{q}_d + C_0(q_d, \dot{q}_d)\dot{q}_d + G_0(q_d)$) is a nominal feed-forward inverse dynamics term intended to minimize the effect of the inherent nonlinearities of the system and hence, to improve the tracking performance.
- The second term (i.e. $A_m r$) consists of a stabilizing state-feedback term specifying the desired closed-loop behavior of the system.
- More importantly, the last term Γ_{ad} , which will be detailed in the subsequent development, is the adaptive signal responsible of rejecting all the remaining disturbances such as friction effects, external disturbances, etc.

Substituting the control law (61) into (29) results in the following error dynamics

$$\dot{r}(t) = A_m r(t) + \Gamma_{ad}(t) - \eta(t, \zeta(t)), \quad r(0) = r_0 \quad (62)$$

where $\eta(t, \zeta(t))$, $\zeta = [r^T, q^T, \dot{q}^T]^T$ is a nonlinear function that gathers all the remaining nonlinearities that result from applying the control law (61) to (29) and is given by

$$\begin{aligned} \eta(t, \zeta(t)) = & M^{-1}(q)(\tilde{M}(q)\ddot{q}_d + \tilde{N}(q, \dot{q})) \\ & + (\mathbb{I} - M^{-1}(q))(A_m r + \Gamma_{AD}) \\ & - \Lambda r + \Lambda^2(q - q_d) \end{aligned} \quad (63)$$

where $\tilde{M}(q) \triangleq M(q) - \hat{M}(q_d)$, $\tilde{N}(q, \dot{q}) \triangleq N(q, \dot{q}) - \hat{N}(q_d, \dot{q}_d)$ and $\mathbb{I} \in \mathbb{R}^{4 \times 4}$ denotes the identity matrix. For the subsequent parametrization of $\eta(t, \zeta(t))$, let's consider the following non-restricting assumptions [41]

Assumption 7. *There exist positive B such that $\|\eta(t, 0)\| \leq B$ holds for all $t \geq 0$.*

Assumption 8. *η is continuous in its arguments, and for arbitrary $\delta > 0$, there exist $d_{\eta_t}(\delta)$ and $d_{\eta_\zeta}(\delta)$ such that*

$$\left\| \frac{\partial \eta(t, \zeta)}{\partial t} \right\|_\infty \leq d_{\eta_t}(\delta), \quad \left\| \frac{\partial \eta(t, \zeta)}{\partial \zeta} \right\|_\infty \leq d_{\eta_\zeta}(\delta) \quad (64)$$

Assumption 9. *for $\tau \geq 0$, $\|r_\tau\|_{\mathcal{L}_\infty} \leq \rho$ and $\|\dot{r}_\tau\|_{\mathcal{L}_\infty} \leq d_r$, for some positive constants ρ and d_r . Now, for some arbitrary $\gamma > 0$, let*

$$\bar{\rho} \triangleq \max \{ \rho + \gamma, L_1(\rho + \gamma) + L_2 \}, \quad L_\rho \triangleq \frac{\bar{\rho}}{\rho} d_{\eta_\zeta}(\bar{\rho}) \quad (65)$$

It follows from Lemma A.9.2 in [41] that $\eta(t, \zeta(t))$ can be parametrized as follows

$$\eta(t, \zeta(t)) = \theta(t) \|r_\tau\|_{\mathcal{L}_\infty} + \sigma(t) \quad (66)$$

where $\theta(t), \sigma(t) \in \mathbb{R}^4$ are differentiable functions. It follows that (62) can be rewritten as

$$\dot{r}(t) = A_m r(t) + \Gamma_{AD}(t) - (\theta(t) \|r_\tau\|_{\mathcal{L}_\infty} + \sigma(t)), \quad r(0) = r_0 \quad (67)$$

Since the nonlinear function $\eta(t, \zeta(t))$ is unknown due to uncertainties and unmeasured disturbances, the control term $\Gamma_{ad}(t)$ should be adaptively designed in order to obtain estimates of $\theta(t)$ and $\sigma(t)$. Now, consider the following state predictor of the combined tracking error

$$\begin{aligned} \dot{\hat{r}}(t) &= A_m \hat{r}(t) + \Gamma_{ad}(t) - \left(\hat{\theta}(t) \|r_\tau\|_{\mathcal{L}_\infty} + \hat{\sigma}(t) \right) \\ &\quad - K \tilde{r}(t), \quad \hat{r}(0) = r_0 \end{aligned} \quad (68)$$

where $\tilde{r}(t) \triangleq \hat{r}(t) - r(t)$ and $K \in \mathbb{R}^{4 \times 4}$ is a design parameter introduced to reject high-frequency noise [14]. $\hat{\theta}(t)$ and $\hat{\sigma}(t)$ are real-time estimates of $\theta(t)$

and $\sigma(t)$ respectively, their evolution is governed by the following projection-based adaptive laws in order to ensure their boundedness [41]

$$\dot{\hat{\theta}}(t) = \Sigma \mathbf{Proj} \left(\hat{\theta}(t), P\tilde{r}(t) \|r_\tau\|_{\mathcal{L}_\infty} \right), \quad \hat{\theta}(0) = \hat{\theta}_0 \quad (69)$$

$$\dot{\hat{\sigma}}(t) = \Sigma \mathbf{Proj} \left(\hat{\sigma}(t), P\tilde{r}(t) \right), \quad \hat{\sigma}(0) = \hat{\sigma}_0 \quad (70)$$

where $\Sigma \in \mathbb{R}^+$ is the adaptive gain, $P = P^T > 0$ is the solution to the algebraic Lyapunov equation $A_m^T P + P A_m = -Q$ for some arbitrary $Q = Q^T > 0$. Since the estimated parameters are bounded thanks to the projection operator, they satisfy $\|\hat{\theta}(t)\|_\infty < \theta_b$, $\|\hat{\sigma}(t)\|_\infty < \sigma_b$, $\forall t \in [0, T]$. The adaptive control signal $\Gamma_{ad}(t)$ is the output of the following system given in Laplace domain

$$\Gamma_{ad}(s) = C(s)\hat{\eta}(s) \quad (71)$$

where $\hat{\eta}(s)$ is the Laplace transform of $\hat{\eta}(t) = \left(\hat{\theta}(t) \|r_t\|_{\mathcal{L}_\infty} + \hat{\sigma}(t) \right)$ and $C(s)$ is a diagonal matrix of filters whose elements are BIBO-stable strictly proper transfer functions satisfying a unity DC gain and zero initialization for their state-space realizations.

7. Real-Time Experiments and Results

In order to demonstrate the relevance of the proposed extended $\mathcal{L}_1$ adaptive controller, real-time experiments were conducted on an experimental testbed consisting of the Veloce robot.

7.1. Experimental Testbed of the VELOCE Robot

The actuators of VELOCE are the TMB0140-100-3RBS ETEL direct-drive motors. They can provide a maximum peak torque of 127 Nm and they are able to reach 550 rpm of speed. Each actuator is equipped with a non-contact incremental optical encoder providing a total number of 5000 pulses per revolution. The global structure of the manipulator is capable of reaching 10 m/s of maximum velocity of the traveling plate, 200 m/s² of maximum acceleration and is able to handle a maximum payload of 10 Kg. The control architecture is implemented using Simulink from Mathworks and compiled using the XPC Target real-time toolbox. The resulting low-level code is then uploaded to the target PC; an industrial computer cadenced at 10 KHz (i.e. sample time of 0.1 ms). The experimental testbed is displayed in Figure 11.

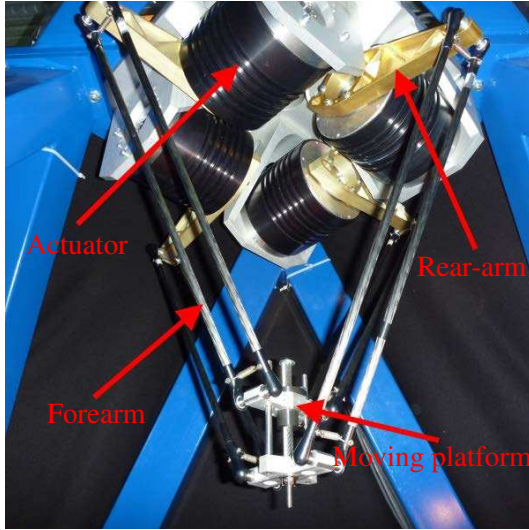


Figure 11. View of the experimental setup of the VELOCE robot.

7.2. Real-Time Experimental Results

To demonstrate the relevance of the proposed contribution and to highlight its benefits, both the $\mathcal{L}_1$ adaptive controller in [16] and the proposed extended one were implemented in real-time on VELOCE. As a reminder, the implemented control law in [16] consists only of the stabilizing term and the adaptive term, it

Table 3. Summary of the controllers' parameters

Parameter	Description	Value
Λ	position error weight	$10 \times \text{diag}(65, 65, 65, 65)$
A_m	transient response matrix	$\text{diag}(-6, -6, -6, -6)$
θ_{max}	upper bound on θ	50
σ_{max}	upper bound on σ	30
Σ	adaptation gain	10^6
K	noise rejection gain	$10^3 \times \text{diag}(6, 6, 6, 6)$
$C(s)$	$\mathcal{L}_1$ low-pass design filter	$144/(s^2 + 21.6s + 144)$

is expressed as

$$\Gamma(t) = A_m r(t) + \Gamma_{ad}(t) \quad (72)$$

The traveling plate of the manipulator had to perform several spatial point-to-point displacements as well as rotations inside the manipulator's workspace. We use 5th order polynomials to generate the desired trajectories in Cartesian space, then the inverse kinematics problem is solved online to determine the corresponding joint trajectories. The duration of each point-to-point trajectory was fixed to 0.2 sec. The desired joint trajectories were obtained by solving the Inverse Kinematics problem in real-time while the actual ones are available from the encoders measurements. The manipulator is not equipped with external sensors, hence the actual Cartesian position is obtained by solving the Forward Kinematics problem in real-time as well. We choose the same control parameters for both controllers, they are summarized in Table 3. The estimated functions were initialized to zero ($\hat{\theta}(0) = [0, 0, 0, 0]^T$, $\hat{\sigma}(0) = [0, 0, 0, 0]^T$).

A comparison of the Cartesian tracking errors between both controllers is depicted in Figure 12. The plots are zoomed in the interval [5, 7] seconds for clarity. It can be clearly seen that the augmented controller performs much better than the original $\mathcal{L}_1$ adaptive controller especially on translations on the x and y -axes, while the enhancement is smaller on z -axis and the rotational DOF. To quantify the improvement brought by the proposed extended $\mathcal{L}_1$ adaptive controller, we formulate the following criteria based on the Root Mean Square (RMS) of the tracking errors:

$$\text{RMS}_J = \left(\sum_{i=1}^4 \text{RMS}^2(\epsilon_{qi}) \right)^{\frac{1}{2}} \quad (73)$$

$$\text{RMS}_C = \left(\text{RMS}^2(\epsilon_x) + \text{RMS}^2(\epsilon_y) + \text{RMS}^2(\epsilon_z) + \text{RMS}^2(\epsilon_s) \right)^{\frac{1}{2}} \quad (74)$$

where $\epsilon_{(\cdot)}$ is the error between the desired and actual position and $\text{RMS}(\cdot)$ is the standard root mean square function. The obtained results are summarized in Table 4. It can be noticed that the improvement is very significant regarding the joint tracking errors (up to 52%) and the translational movements (up to 68.8%) while only small improvement is observed on the rotation of the traveling plate (only 4.2%). This result highlights the benefits of using the dynamics of the manipulator in the control loop in terms of tracking performance.

The generated control inputs are shown in Figure 15. The control inputs remain within the allowable range and do not exceed the limits of the actuators.

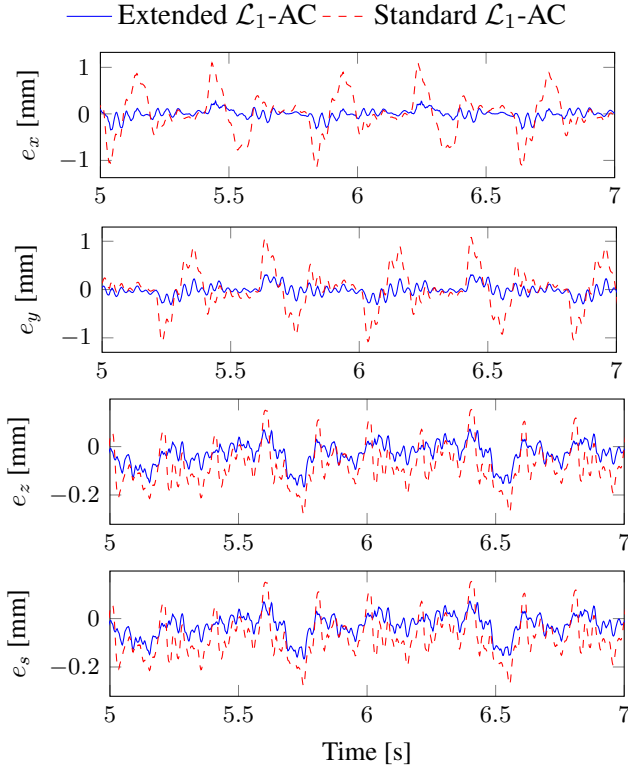


Figure 12. Cartesian tracking errors: Extended $\mathcal{L}_1$ adaptive controller (solid line), $\mathcal{L}_1$ adaptive controller (dashed line).

Overall, it can be observed that the amplitudes of the control inputs in the case of the augmented controller are slightly smaller than those of the standard $\mathcal{L}_1$ adaptive one. Hence, the following conclusion can be drawn: the proposed

Table 4. Tracking performance comparison

	$\mathcal{L}_1$ -AC	Extended $\mathcal{L}_1$ -AC	Improvement
RMS_J [deg]	0.1012	0.0486	52 %
RMS_C [mm]	0.4634	0.195	69.7 %

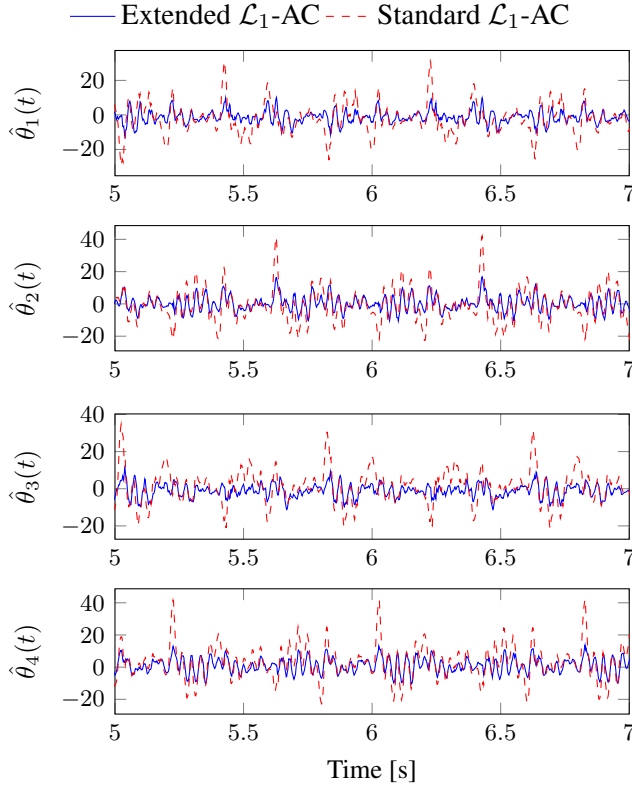


Figure 13. Evolution of $\hat{\theta}$ versus time.

controller significantly improves the tracking performance of the closed-loop system while consuming less energy than the standard one.

The evolution of the estimated nonlinear functions $\hat{\theta}(t)$ and $\hat{\sigma}(t)$ versus time is depicted in Figure 13 and Figure 14 respectively. These figures clearly demonstrate the relevance of the proposed contribution. Indeed, the addition of the model-based feedforward considerably helps in compensating for the modeling inherent nonlinearities of the manipulator. Therefore, the remaining nonlinearities that have to be compensated for by the adaptive signal are of smaller amplitudes. This result is further illustrated in Figure 16 where only the evolution of the adaptive component of (61) and (72) are plotted (i.e $\Gamma_{ad}(t)$). We can see that the adaptive signal, needed to compensate for the remaining nonlinearities, of the proposed controller is of lower amplitude than that of the

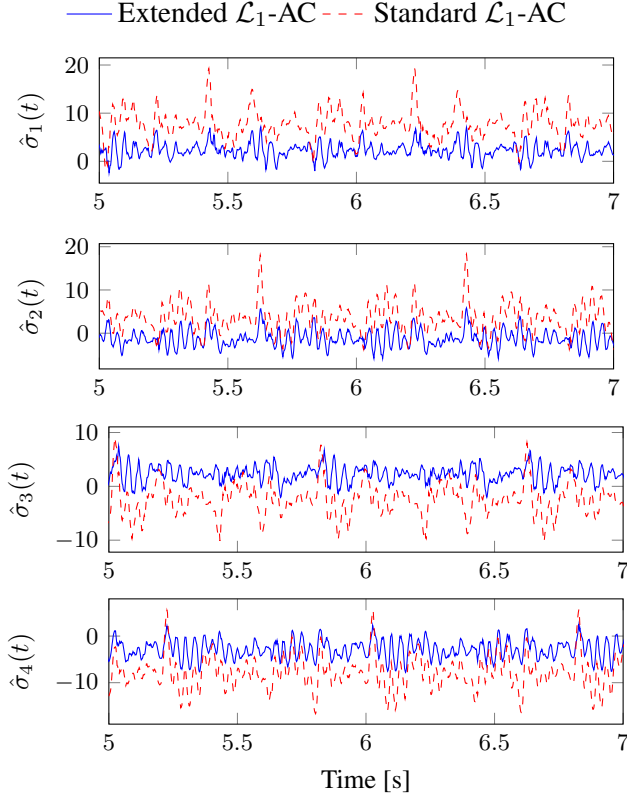


Figure 14. Evolution of $\hat{\sigma}$ versus time.

conventional one.

Conclusion

In this work, a new controller based on $\mathcal{L}_1$ adaptive control for parallel manipulators is presented. Standard $\mathcal{L}_1$ adaptive control do not rely on any knowledge about the dynamics of the controlled system which may result in a poor tracking performance. Since a simplified dynamic model for parallel manipulators is relatively easy to derive, we propose to take advantage of the modelled dynamics in the control loop to enhance the performance of $\mathcal{L}_1$ adaptive control. For this reason, we augment the standard $\mathcal{L}_1$ adaptive controller with a computed

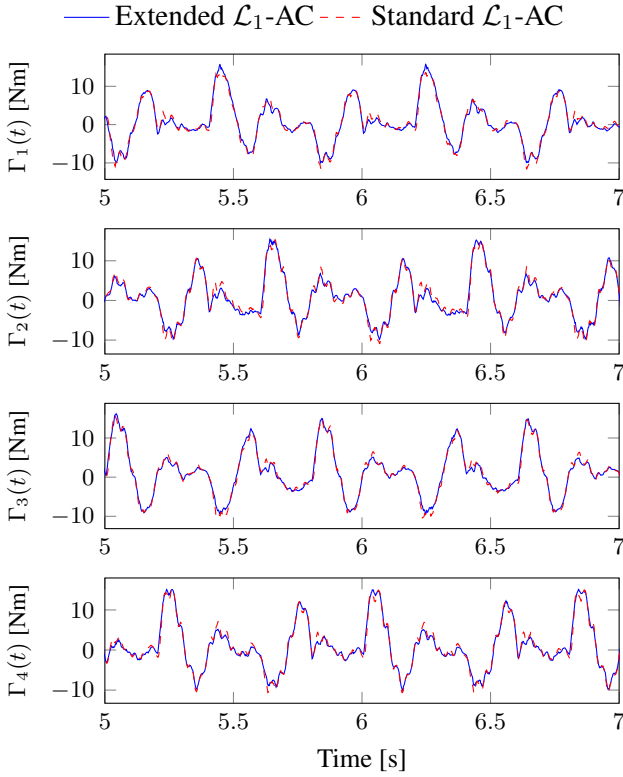


Figure 15. Evolution of the control input torques Γ versus time.

feedforward term based on the nominal dynamics of the parallel manipulator and the desired trajectories. The main motivation behind such a proposition is to improve the tracking performance of the manipulator. In fact, the included dynamics partially compensates for the inherent nonlinear dynamics which results in an improved tracking performance. Real-time experiments on a 4-DOF parallel manipulator show that the proposed augmented control scheme significantly reduces the tracking errors as compared to standard $\mathcal{L}_1$ adaptive control. Future work may consider evaluating the proposed controller in other working scenarios and on other parallel manipulators prototypes.

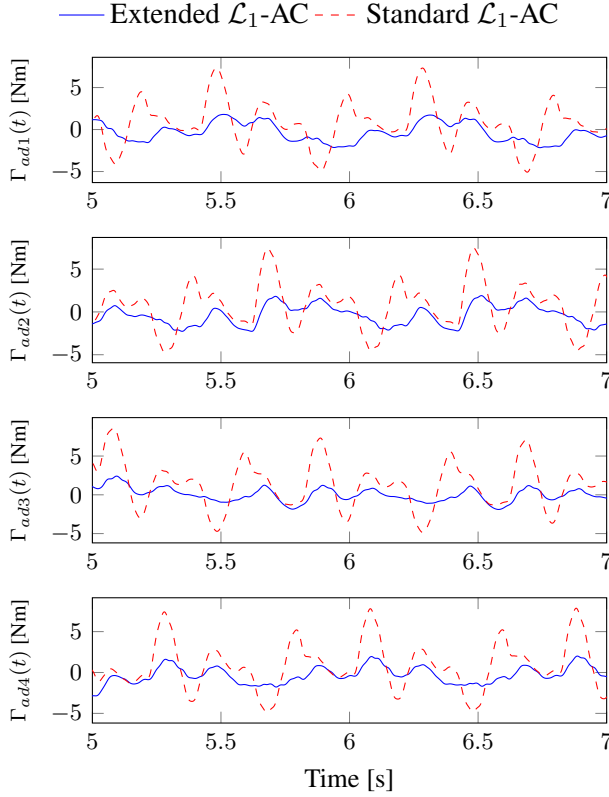


Figure 16. Evolution of the adaptive control input torques Γ_{ad} versus time.

Acknowledgments

This research was supported by the French National Research Agency, within the project ARROW (ANR-2011-BS3-006-01-ARROW).

References

- [1] J. Luh, M. Walker, and R. Paul, “Resolved-acceleration control of mechanical manipulators,” *IEEE Trans. Autom. Control*, vol. 25, no. 3, pp. 468–474, Jun. 1980.

-
- [2] D. Koditschek, "Natural motion for robot arms," in *Proc. The 23rd IEEE Conference on Decision and Control (CDC'84)*, Las Vegas, Nevada, USA, Dec. 1984, pp. 733–735.
 - [3] B. Paden and R. Panja, "Globally asymptotically stable pd+ controller for robot manipulators," *International Journal of Control*, vol. 47, no. 6, pp. 1697–1712, Jun. 1988.
 - [4] S. Dubowsky and D. DesForges, "The application of model-referenced adaptive control to robotic manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 101, no. 3, pp. 193–200, Jun. 1979.
 - [5] R. Horowitz and M. Tomizuka, "An adaptive control scheme for mechanical manipulators - compensation of nonlinearity and decoupling control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 108, no. 2, pp. 127–135, Jun. 1986.
 - [6] J. Craig, P. Hsu, and S. Sastry, "Adaptive control of mechanical manipulators," *The International Journal of Robotics Research*, vol. 6, no. 2, pp. 16–28, Jun. 1987.
 - [7] R. Ortega and M. W. Spong, "Adaptive motion control of rigid robots: A tutorial," *Automatica*, vol. 25, no. 6, pp. 877–888, Oct. 1989.
 - [8] N. Sadegh and R. Horowitz, "Stability and robustness analysis of a class of adaptive controllers for robotic manipulators," *The International Journal of Robotics Research*, vol. 9, no. 3, pp. 74–92, Jun. 1990.
 - [9] M. Bennehar, A. Chemori, and F. Pierrot, "A new extension of direct compensation adaptive control and its real-time application to redundantly actuated pkms," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'14)*, Chicago, Illinois, USA, Sep. 2014, pp. 1670–1675.
 - [10] C. Cao and N. Hovakimyan, "Design and analysis of a novel $\mathcal{L}_1$ adaptive controller, part i: Control signal and asymptotic stability," in *American Control Conference (ACC'06)*, Minneapolis, USA, Jun. 2006, pp. 3397–3402.

- [11] —, “Design and analysis of a novel $\mathcal{L}_1$ adaptive controller, part ii: Guaranteed transient performance,” in *American Control Conference (ACC’06)*, Minneapolis, USA, Jun. 2006, pp. 3403–3408.
- [12] I. Gregory, C. Cao, E. Xargay, N. Hovakimyan, and X. Zou, “L1 adaptive control design for nasa airstar flight test vehicle,” *AIAA Guidance, Navigation, and Control Conference*, p. 5738, Aug. 2009.
- [13] D. Maalouf, V. Creuze, and A. Chemori, “A novel application of multi-variable $\mathcal{L}_1$ adaptive control: From design to real-time implementation on an underwater vehicle,” in *Proc. IEEE International Conference on Intelligent Robots and Systems (IROS’12)*, Vilamoura, Algarve, Portugal, Jan. 2012, pp. 76–81.
- [14] K. D. Nguyen, H. Dankowicz, and N. Hovakimyan, “Marginal stability in $\mathcal{L}_1$ adaptive control of manipulators,” in *Proceedings of the 9th ASME International Conference on Multibody Systems, Nonlinear Dynamics, and Control*, Portland, OR, USA, 2013.
- [15] L. Techy, C. K. Reddy, C. A. Woolsey, C. Cao, and N. Hovakimyan, “Non-linear control of a novel two-link pendulum,” in *American Control Conference (ACC’07)*, New York, USA, 2007, pp. 19–24.
- [16] M. Bennehar, A. Chemori, and F. Pierrot, “L1 adaptive control of parallel kinematic manipulators: Design and real-time experiments,” in *IEEE International Conference on Robotics and Automation (ICRA’15)*, Seattle, Washington, USA, May 2015, pp. 1587–1592.
- [17] K. Astrom, *Adaptive Control (Second Edition)*. Addison-Wesley, 1995.
- [18] C. C. Nguyen, S. S. Antrazi, Z. L. Zhou, and C. E. Campbell, “Adaptive control of a stewart platform-based manipulator,” *Journal of Robotic Systems*, vol. 10, no. 5, pp. 657–687, 1993.
- [19] S. Dubowsky and D. DesForges, “The application of model-referenced adaptive control to robotic manipulators,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 101, no. 3, pp. 193–200, Sep. 1979.

-
- [20] S. Nicosia and P. Tomei, "Model reference adaptive control algorithms for industrial robots," *Automatica*, vol. 20, no. 5, pp. 635 – 644, 1984.
 - [21] B. Yegnanarayana, *Artificial neural networks*. PHI Learning Pvt. Ltd., 2009.
 - [22] X. Qingsong and L. Yangmin, "A 3-prs parallel manipulator control based on neural network," in *Advances in Neural Networks ISNN 2007*, D. Liu, S. Fei, Z.-G. Hou, H. Zhang, and C. Sun, Eds. Springer Berlin Heidelberg, 2007, pp. 757–766.
 - [23] F. Lewis, S. Jagannathan, and A. Yesildirak, *Neural network control of robot manipulators and non-linear systems*. CRC Press, 1998.
 - [24] Y. Li and Y. Wang, "Trajectory tracking control of a redundantly actuated parallel robot using diagonal recurrent neural network," in *International Conference on Natural Computation (ICNC'09)*, vol. 2, Aug 2009, pp. 292–296.
 - [25] J. J. Craig, P. Hsu, and S. S. Sastry, "Adaptive control of mechanical manipulators," *The International Journal of Robotics Research*, vol. 6, no. 2, pp. 16–28, 1987.
 - [26] H. Cheng, Y. Yiu, and Z. Li, "Dynamics and control of redundantly actuated parallel manipulators," *IEEE/ASME Transactions on Mechatronics*, vol. 8, no. 4, pp. 483–491, Dec. 2003.
 - [27] J. Merlet, *Parallel Robots, Second Edition*. Dordrecht, Netherlands: Springer, 2006.
 - [28] W. Shang, S. Cong, and Y. Ge, "Adaptive computed torque control for a parallel manipulator with redundant actuation," *Robotica*, vol. 30, no. 3, pp. 457–466, May 2012.
 - [29] M. Honegger, R. Brega, and G. Schweiter, "Application of a nonlinear adaptive controller to a 6 dof parallel manipulator," in *IEEE International Conference on Robotics and Automation (ICRA'00)*, vol. 2, San Francisco, CA, USA, Apr. 2000, pp. 1930–1935.

- [30] G. Sartori Natal, A. Chemori, and F. Pierrot, "Dual-space control of extremely fast parallel manipulators: Payload changes and the 100g experiment," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1520–1535, Jul. 2015.
- [31] D. Corbel, M. Gouttefarde, O. Company, and F. Pierrot, "Actuation redundancy as a way to improve the acceleration capabilities of 3t and 3t1r pick-and-place parallel manipulators," *Journal of Mechanisms and Robotics*, vol. 2, no. 4, pp. 1–13, 2010.
- [32] W. Shang and S. Cong, "Nonlinear adaptive task space control for a 2-dof redundantly actuated parallel manipulator," *Nonlinear Dynamics*, vol. 59, no. 1-2, pp. 61–72, 2010.
- [33] J.-J. Slotine and L. Weiping, "On the adaptive control of robot manipulators," *The International Journal of Robotics Research*, vol. 6, no. 3, pp. 49–59, Sep. 1987.
- [34] R. Clavel, "Device for the movement and positioning of an element in space," Dec. 1985, uS Patent 4,976,582. [Online]. Available: <https://patents.google.com/patent/US4976582A>
- [35] J. Merlet, *Parallel Robots, Second Edition*. Dordrecht, Netherlands: Springer, 2006.
- [36] A. Codourey, "Dynamic modeling of parallel robots for computed-torque control implementation," *The International Journal of Robotics Research*, vol. 17, pp. 1325–1336, 1998.
- [37] D. Corbel, M. Gouttefarde, and O. Company, "Towards 100g with pkm. is actuation redundancy a good solution for pick and place?" in *Proc. IEEE International Conference on Robotics and Automation (ICRA'10)*, Anchorage, Alaska, May 2010, pp. 4675–4682.
- [38] N. Hovakimyan and C. Cao, *L1 Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation*. Philadelphia, Pennsylvania, USA: SIAM, 2010.
- [39] M. Bennehar, A. Chemori, and F. Pierrot, "L1 adaptive control of parallel kinematic manipulators: Design and real-time experiments," in *IEEE*

International Conference on Robotics and Automation (ICRA'15), Seattle, Washington, USA, May 2015, pp. 1587–1592.

- [40] H. Khalil, *Nonlinear Systems*. Prentice-Hall, 2002.
- [41] N. Hovakimyan and C. Cao, *L1 Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation*. Philadelphia, Pennsylvania, USA: SIAM, 2010.

Chapter 2

A REDUNDANT DYNAMIC MODELLING PROCEDURE BASED ON EXTRA SENSORS FOR PARALLEL ROBOT CONTROL

Asier Zubizarreta, Itziar Cabanes, Marga Marcos,
Charles Pinto, Javier Corral and Pablo Bengoa*

Faculty of Engineering in Bilbao,
University of the Basque Country (UPV/EHU)
Bilbao, Spain

Keywords: parallel robots, Gough-Stewart platform, position control, computed torque control, modelling

1. Introduction

Robots are a key element in current industrial processes, as they can be applied to a number of tasks, increasing both quality and productivity. Traditionally, serial robots have been installed in factories, as their wide operating space allowed them to fulfill a number of tasks. However, due to their high moving mass and single kinematic chain structure, these robots present some disadvantages when high speed, accuracy or heavy load handling tasks have to be executed.

*Corresponding Author: asier.zubizarreta@ehu.es

Parallel robots [27] provide an interesting alternative to these application fields, as their multiple kinematic chain structure offers increased stiffness, allowing reduced positioning errors, lighter mechanisms and increased load/weight ratios. These capabilities make these robots suitable for those tasks where high accelerations and speeds, micrometric precision or heavy load handling are required.

However, the complexity of their structure presents some drawbacks: complex kinematic and dynamic modelling, lack of closed-form solution in the kinematic problem, presence of nonactuated joints, orientation and position coupling, limbs interference, small workspace and presence of inner singularities. These issues limit an extensive implementation of these robots in practical applications in industry. Moreover, being a recently rediscovered field, there are still many areas in parallel robotics that have not been thoroughly studied. Thus, in addition to the aforementioned problematic, other challenges in areas such as control or calibration arise.

The present work focuses on the control of parallel robots. Although less studied than areas such as kinematic analysis or synthesis, control plays an important role in parallel robots [26], as it allows to extract the dynamic capabilities of the mechanical structure, ensuring that the design specifications are met in real robots. Moreover, parallel robots are usually implemented in tasks with high performance requirements where advanced control laws are required to fulfill the required specifications. Hence, research in this area is a central issue in parallel robotics.

However, most of control approaches in parallel robotics have been imported from serial robots without considering the particularities of these robots. One of the main issues is that while control laws are easily defined in the joint-space in serial robots, in parallel robots the natural coordinates are the ones defined in task space [31]. These, which represent the pose of the mobile platform are not easily measured directly in the general case. Thus, it is common to use estimators based on the kinematic model of the robot, whose resolution is based on numerical procedures such as Newton-Raphson iterative approach. Moreover, in parallel robots some joints are not actuated, and their motion also has to be estimated.

In this work, a solution based on the use of extra sensors attached to the unactuated joints of the parallel robot is detailed. The extra information provided by these sensors can be used to implement a model-based control approach that combines the performance of advanced control techniques and the robustness of

sensor redundancy. In order to implement this approach, three major areas will be analyzed in the next sections: dynamic modelling, control law definition, and sensitivity, considering the well known Gough-Stewart platform as study case.

2. Modelling Methodology Based on Extra Sensors

2.1. Definition of the Dynamic Model in Parallel Robots

In serial robotics literature, Lagrangian and Newton-Euler formulations have been widely used to calculate the dynamic model of n degrees-of-freedom serial robots [40].

$$\boldsymbol{\tau} = \mathbf{D}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) \quad (1)$$

where $\boldsymbol{\tau}$ is the vector of the actuator torque/forces, $\mathbf{q}$ is the vector of the n articular coordinates, $\mathbf{D}$ is the symmetric and positive-defined inertia matrix, $\mathbf{C}$ is the Coriolis matrix, where $(\mathbf{D} - 2\mathbf{C})$ is skew-symmetric and $\mathbf{G}$ is the gravity terms vector. Friction terms have been neglected in this model, although they can be introduced in it as an additional term easily. This model has been used in the literature to implement multiarticular model-based control techniques such as the Computed Torque Control approach [9].

The dynamic model of parallel robots can also be defined in the joint space as in Eq. (1), although some of the properties of the dynamic model do not hold, as the dynamic model is not defined in Direct Kinematic Problem singularities [23]. Being more complex than serial robots, the procedures applied to serial robots cannot be used in parallel robots directly, as constraints due to the multiple kinematic chains have to be introduced in the dynamic model.

In the literature, three main approaches have been proposed to obtain the dynamic model of parallel robots: the traditional Newton-Euler formulation [10, 14, 18], which usually leads to a large number of equations due to the internal forces of the mechanism; the Lagrangian formulation using Lagrange multipliers [5, 20], which is an energetic approach and allows to eliminate the need of internal forces calculation; and the Principle of Virtual Work [12, 41, 43], which is also an energetic approach. When modelling a particular parallel robot, any approach can lead to a dynamic model as the one described in Eq. (1).

However, these approaches require the knowledge of all kinematic variables of the robot in order to calculate the dynamic model of parallel robots. As the measurement of the end effector pose, this is, the Tool Center Point location, is not possible in the general case, the required kinematic variables need to be

estimated in terms of the actuated joints, which are usually sensorized. This approach usually requires iterative procedures to calculate the estimated variables, and robustness of the estimation is based on the accuracy of the kinematic model parameters.

In order to provide extra robustness to the approach and reduce computational time, extra sensors can be introduced in specific passive joints. The advantages of the use of redundant data have been demonstrated in several works [3,25] to improve estimation accuracy or decrease kinematic model computational time. In this work, the use of explicit extra data in the dynamic model is proposed, so that this extra information can be used to increase control performance in model-based control approaches. This way, in the next section, an extended procedure is detailed to define the dynamic model of a parallel robot in terms of whatever generalized coordinates the designer selects, allowing to introduce data from extra sensors directly in the dynamic model.

2.2. Dynamic Modelling Methodology for Full Parallel Robots

The proposed methodology is based in two main ideas: the use of the Lagrangian Formulation, which allows to define the dynamic model in terms of any set of generalized variables; and the use of a subsystem approach, in which the parallel robot is divided in the moving platform and the limbs, allowing to calculate the model of each part easily. It should be noted that the proposed approach is valid for fully parallel robots, i.e., robots in which the number of limbs are equal to the number of the degrees of freedom of the robot. These robots are very common in parallel robotic applications.

In addition, the following considerations apply, which are common in general application cases: nondeformable elements are considered and friction terms will be neglected, although their introduction is straightforward.

In order to illustrate the approach, the Gough-Stewart platform will be considered as study case (Fig. 1). The Gough-Stewart platform is one of the most studied parallel robots, being composed by a moving platform and six *UPS* serial chains. This structure provides 6 degrees-of-freedom, making it suitable for tasks such as heavy load handling [37,44], high precision [28], underwater robotics [35], nuclear facilities inspection [34] and motion simulation [22].

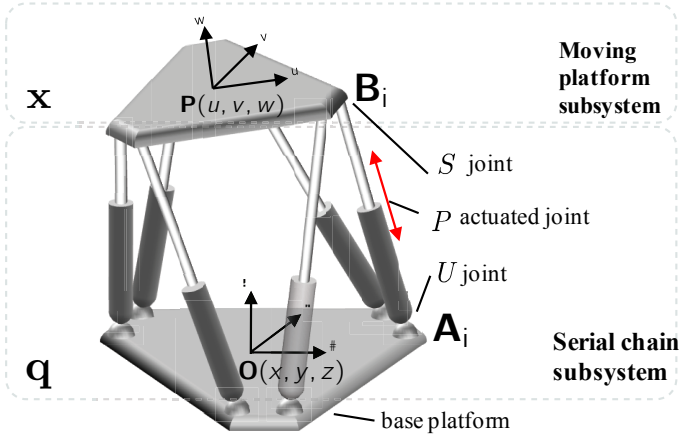


Figure 1. Gough-Stewart platform.

2.2.1. Previous Considerations: Orientation Angles and Angular Velocities

In the general case, for a 6 degrees-of-freedom motion, the output or task coordinates $\mathbf{x}$ contain the cartesian position of the Tool Center Point (TCP) $\mathbf{p}_x = [x \ y \ z]^T$, and the orientation angles $\psi = [\alpha \ \beta \ \gamma]^T$ of the moving platform with respect to the fixed frame $\mathbf{O}(x, y, z)$,

$$\mathbf{x} = [x \ y \ z \ \alpha \ \beta \ \gamma]^T \quad (2)$$

However, the choice of the orientation angles is arbitrary, and depends on the particular convention used. Hence, the time derivative of $\mathbf{x}$ only has physical meaning for the translation terms [2] in the general case, as the derivative of the orientation angles $\dot{\psi}$ is not, in the general case, the angular velocity of the moving platform, $\dot{\psi} \neq \omega$.

Hence, as the dynamics of rotational motions are described in terms of the angular velocity ω but traditionally orientation is defined in terms of Euler angles ψ , both variables should be related if a general case is to be considered. This relationship can be derived as [9],

$$\tilde{\omega} = \dot{\mathbf{R}} \mathbf{R}^T \quad (3)$$

where $\mathbf{R}(\psi)$ is the rotation matrix of the moving platform with respect to the fixed frame, $\dot{\mathbf{R}}(\psi, \dot{\psi})$ is its time derivative and $\tilde{\omega}$ is a skew-symmetric matrix

related to ω ,

$$\tilde{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad \omega = [\omega_x \ \omega_y \ \omega_z]^T \quad (4)$$

Eq. 3 can be rewritten in the following form, depending on the selected Euler angle convention,

$$\omega = \mathbf{E} \dot{\psi} = \mathbf{E} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \quad (5)$$

where $\mathbf{E}$ depends on the selected Euler angle convention, and will be illustrated for a specific case in the next subsection.

Hence, if a velocity vector $\mathbf{v}$ is defined in terms of the angular velocity terms,

$$\mathbf{v} = \begin{bmatrix} \dot{\mathbf{p}}_x \\ \omega \end{bmatrix} \quad (6)$$

then, $\dot{\mathbf{x}}$ and $\mathbf{v}$ can be related as,

$$\mathbf{v} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{E} \end{bmatrix} \dot{\mathbf{x}} = \mathbf{T}_R \dot{\mathbf{x}} \quad (7)$$

where $\mathbf{T}_R(\psi)$ is a projection matrix defined in terms of the selected Euler angles.

Based on this expression, and taking the time derivative, the angular acceleration and the second time derivatives of Euler angles can be related,

$$\mathbf{a} = \dot{\mathbf{v}} = \begin{bmatrix} \ddot{\mathbf{p}}_x \\ \dot{\omega} \end{bmatrix} = \mathbf{T}_R \ddot{\mathbf{x}} + \dot{\mathbf{T}}_R \dot{\mathbf{x}} \quad (8)$$

where $\dot{\mathbf{T}}_R(\psi, \dot{\psi})$ depends on the Euler angles and their derivatives.

Eqs. 7 and 8 will be used in the next section to define the dynamic model of the robot.

2.2.2. Subsystems and Variable Sets

A parallel robot is composed by two platforms, a fixed one and a moving one, connected by several serial links (Fig. 1). If a fully parallel robot is considered,

such as the Gough-Stewart platform, the number of serial chains or limbs is equal to the number of degrees-of-freedom (dof).

In order to calculate the dynamic model of the robot, the usual approach is to consider the parallel robot as a group of independent subsystems that are connected using kinematic constraints. Thus, two subsystems are considered: the moving platform, where the TCP is located, and the serial chain subsystem, i.e, the limbs. The dynamic model of each subsystem can be calculated easily in terms of its natural coordinates. This way, the motion of the moving platform can be easily defined in the task space in terms of the $n = 6$ output coordinates $\mathbf{x} = [x \ y \ z \ \alpha \ \beta \ \gamma]^T$, while the dynamics of the serial limbs can be calculated using serial robot approaches, which are based on the use of $n_q = 18$ joint variables $\mathbf{q}$.

Joint variables can be divided in two sets: actuated and nonactuated joint variables,

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_a \\ \mathbf{q}_{na} \end{bmatrix} \in \mathbb{R}^{18} \quad (9)$$

The $n = 6$ actuated joints $\mathbf{q}_a$, are associated to the prismatic actuators of the Gough-Stewart platform, which are composed by a cylinder and a rod, and whose relative motion varies the $q_{a_i} = l_i$ distance between points $\mathbf{A}_i$ and $\mathbf{B}_i$ for each $i = 1, \dots, 6$ leg (Fig. 2).

$$\mathbf{q}_a = [l_1 \ l_2 \ l_3 \ l_4 \ l_5 \ l_6]^T \quad (10)$$

Nonactuated joint variables $\mathbf{q}_{na}$, on the other hand, are not actuated although their motion is required to be known in order to calculate the dynamic model of the robot (Fig. 2),

$$\mathbf{q}_{na} = \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\varphi} \end{bmatrix} \in \mathbb{R}^{12} \quad (11)$$

where, $\boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4 \ \theta_5 \ \theta_6]^T$ and $\boldsymbol{\varphi} = [\varphi_1 \ \varphi_2 \ \varphi_3 \ \varphi_4 \ \varphi_5 \ \varphi_6]^T$.

In this work, the introduction of sensors in a subset of the nonactuated joints is considered. Hence, the joint variables in the nonactuated joint variable set can be divided in n_s sensorized $\mathbf{q}_s$ and n_p passive $\mathbf{q}_p$ joint variables, $\mathbf{q}_{na} = [\mathbf{q}_s^T \ \mathbf{q}_p^T]^T$. This way, the set of sensorized and measurable $n_c = n + n_s$

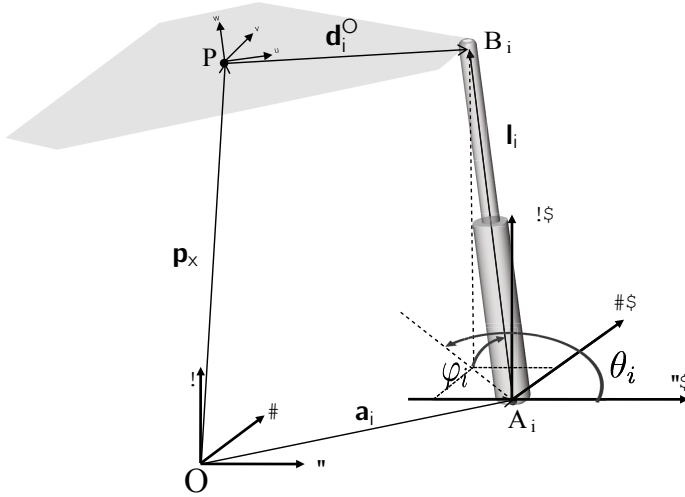


Figure 2. Closed loop kinematic equation.

joints will be denoted as *control coordinates* $\mathbf{q}_c$ (Fig.3),

$$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a \\ \mathbf{q}_s \end{bmatrix} \in \mathbb{R}^{n_c}$$

where $n_c \in [n, n_q]$, depending on the number of extra sensors introduced to measure the variables in $\mathbf{q}_{na}$.

2.2.3. Kinematic Model

In order to define the dynamic model of the Gough-Stewart platform, its kinematic relations have to be calculated first.

Closure Loop Equation. The six vectorial closure loop equations are deduced from the mechanical structure of each limb and its connection to the fixed and moving platform, as seen in Fig. 2. This way, if $O(x, y, z)$ is the fixed reference frame, attached to the fixed platform, the equations defining the motion of each limb can be written as follows,

$$\Gamma_i = \mathbf{a}_i + \mathbf{l}_i - \mathbf{p}_x - \mathbf{d}_i^O = \mathbf{0} \quad i = 1, \dots, 6 \quad (12)$$

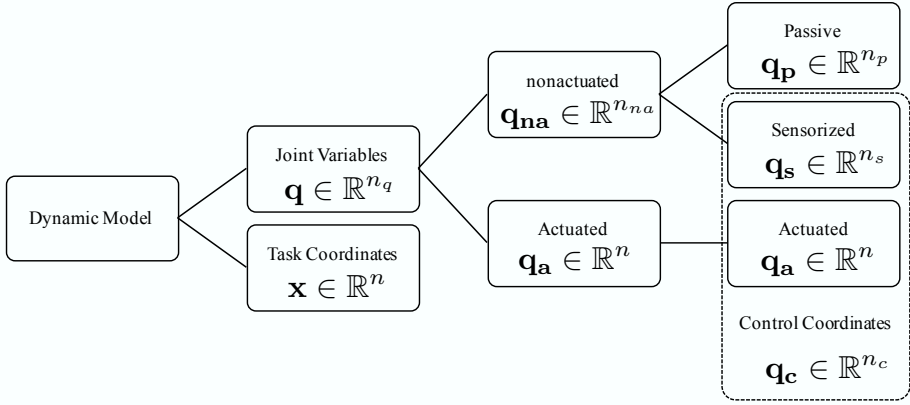


Figure 3. Variables in the dynamic modelling methodology.

where $\mathbf{p}_x = [x \ y \ z]^T$ is the cartesian position of the TCP of the robot, which is the origin of the moving frame; $\mathbf{a}_i$ is the constant position vector of the U joints located in $\mathbf{B}_i$, which define the geometry of the fixed platform and $\mathbf{l}_i$ is the vector that represents the relative position of $\mathbf{A}_i$ with respect to $\mathbf{B}_i$, and it is defined in terms of the joint variables associated to the limbs $\mathbf{q}_i = [l_i \ \theta_i \ \varphi_i]^T$,

$$\mathbf{l}_i = l_i \begin{bmatrix} \cos \theta_i \cos \varphi_i \\ \sin \theta_i \cos \varphi_i \\ \sin \varphi_i \end{bmatrix} \quad (13)$$

The orientation of the moving platform is given by $\mathbf{d}_i^O$, which is the projection of the constant vector $\mathbf{d}_i$ defined in the moving reference frame $P(u, v, w)$ with respect to the fixed one,

$$\mathbf{d}_i^O = \mathbf{R}(\alpha, \beta, \gamma) \mathbf{d}_i \quad (14)$$

where if roll-pitch-yaw (α, β, γ) angles are selected to define the orientation,

$$\mathbf{R}(\alpha, \beta, \gamma) = \begin{bmatrix} c\gamma c\beta & c\gamma s\beta s\alpha - s\gamma c\alpha & s\gamma s\alpha + c\gamma s\beta c\alpha \\ s\gamma c\beta & c\gamma c\alpha + s\gamma s\beta s\alpha & s\gamma s\beta c\alpha - c\gamma s\alpha \\ -s\beta & c\beta s\alpha & c\beta c\alpha \end{bmatrix} \quad (15)$$

where c and s stand for the trigonometric functions cosine and sine, respectively.

Note that each closure loop equation is defined using two sets of variables, the output ones $\mathbf{x}$, and the joint variables associated to each limb $\mathbf{q}_i$. The combination of the closure loop equations 12, defines the constraint equation set of the robot, that relates all variables,

$$[\Gamma_i(\mathbf{x}, \mathbf{q}_i)]_{i=1,\dots,6} = \Gamma(\mathbf{x}, \mathbf{q}) = \mathbf{0}_{18 \times 1} \quad (16)$$

Inverse Kinematic Problem. The Inverse Kinematic Problem (IKP) allows the calculation of the active joint variables $\mathbf{q}_a$ in terms of the output ones $\mathbf{x}$. Thus, if $\mathbf{x}$ is known, the cartesian position of $\mathbf{B}_i = \mathbf{p}_x + \mathbf{R} \mathbf{d}_i$ can be determined easily. Hence, imposing the distance constraint $l_i^2 = \|\mathbf{l}_i\|^2$ on the vectorial loop closure equation Eq.12 and operating,

$$l_i^2 = \mathbf{l}_i^T \mathbf{l}_i = [\mathbf{p}_x - \mathbf{a}_i + \mathbf{R}(\alpha, \beta, \gamma) \mathbf{d}_i]^T [\mathbf{p}_x - \mathbf{a}_i + \mathbf{R}(\alpha, \beta, \gamma) \mathbf{d}_i] \quad i = 1 \dots 6 \quad (17)$$

Position Problem for Nonactuated Joints. The nonactuated joint variables define the position of the elements of each serial chain. Hence, if vector $\mathbf{l}_i$ associated to each limb is considered and Eqs. 12 and 13 are combined,

$$\mathbf{l}_i = \mathbf{p}_x - \mathbf{a}_i + R(\alpha, \beta, \gamma) \mathbf{d}_i = \begin{bmatrix} l_i \cos \theta_i \cos \varphi_i \\ l_i \sin \theta_i \cos \varphi_i \\ l_i \sin \varphi_i \end{bmatrix} \quad (18)$$

Considering $\mathbf{q}_a$ and $\mathbf{x}$ known, each nonactuated joint variable can be calculated for $i = 1, \dots, 6$,

$$\varphi_i = \text{atan2} \left(l_{iz}, \sqrt{l_{ix}^2 + l_{iy}^2} \right) \quad \theta_i = \text{atan2}(l_{iy}, l_{ix}) \quad (19)$$

Direct Kinematic Problem. The Direct Kinematic Problem (DKP) calculates the pose of the moving platform $\mathbf{x}$ in terms of the actuated ones $\mathbf{q}_a$, whose actuators are usually sensorized. This relationship is of great importance in robotics, as $\mathbf{x}$ cannot be directly measured in the general case, and its estimation is required. However, this estimation is usually time consuming, as no analytical solution exists in the general case for the DKP. Moreover, the estimation provided by the DKP is highly dependent on the accuracy of the kinematic parameters of the robot.

Redundant sensors can be used to enhance the calculation procedure of the DKP [25, 32] and add more robustness. This way, if a subset of the nonactuated

joints are sensorized, $\mathbf{q}_s$, this data can be used to get a better estimation of the output variables.

For this purpose, a selection vector $\mathbf{I}_c$ is defined. This vector contains the indexes of the variables of $\mathbf{q}$ that are sensorized, considering that active ones $\mathbf{q}_a$ are always in this set. Hence, the set of control variables $\mathbf{q}_c$ can be defined,

$$\mathbf{q}_c = [\mathbf{q}]_{\mathbf{I}_c} \quad (20)$$

Hence, two sets of equations can be used to estimate $\mathbf{x}$. First, if Eq.17 is considered, $\mathbf{x}$ and $\mathbf{q}_a$ can be related,

$$[||\mathbf{l}_i(\mathbf{x})|| - l_i^2]_{i=1\dots 6} = \mathbf{0}_{6 \times 1} \quad (21)$$

On the other hand, the sensorized nonactuated joint variables defined in Eq. 19 can be also used to estimate $\mathbf{x}$,

$$\begin{aligned} [l_{ix}(\mathbf{x}, \mathbf{q}_a) \tan \theta_i - l_{iy}(\mathbf{x}, \mathbf{q}_a)]_{i=1\dots 6} &= \mathbf{0}_{6 \times 1} \\ \left[\sqrt{l_{ix}(\mathbf{x}, \mathbf{q}_a)^2 + l_{iy}(\mathbf{x}, \mathbf{q}_a)^2} \tan \varphi_i - l_{iz}(\mathbf{x}, \mathbf{q}_a) \right]_{i=1\dots 6} &= \mathbf{0}_{6 \times 1} \end{aligned} \quad (22)$$

Combining Eqs. 21 and 22, a 18 equation system can be defined, each relating a variable from $\mathbf{q}$ with the output variables, $\mathbf{F} = \mathbf{0}_{18 \times 1}$,

$$\mathbf{F}(\mathbf{x}, \mathbf{q}) = \begin{bmatrix} [||\mathbf{l}_i|| - l_i^2]_{i=1\dots 6} \\ [l_{ix} \tan \theta_i - l_{iy}]_{i=1\dots 6} \\ \left[\sqrt{l_{ix}^2 + l_{iy}^2} \tan \varphi_i - l_{iz} \right]_{i=1\dots 6} \end{bmatrix} = \mathbf{0}_{18 \times 1} \quad (23)$$

Hence, selecting the sensorized variables using $\mathbf{I}_c$, the equation system relation all measurable variables $\mathbf{q}_c$ and $\mathbf{x}$ can be obtained,

$$\mathbf{F}_c(\mathbf{q}_c, \mathbf{x}) = [\mathbf{F}(\mathbf{q}, \mathbf{x})]_{\mathbf{I}_c} = \mathbf{0} \quad (24)$$

which, as $\mathbf{q}_c$ is known, defines the redundant DKP solving approach,

Velocity Problem. In order to calculate the velocity relations of the robot, the vectorial Closure Loop equation for each limb Eq. 12 has to be differentiated with respect to time,

$$\dot{\mathbf{l}}_i = \dot{\mathbf{p}}_x + \dot{\mathbf{R}} \mathbf{d}_i = \dot{\mathbf{p}}_x + \boldsymbol{\omega} \times \mathbf{R} \mathbf{d}_i = \dot{\mathbf{p}}_x - \mathbf{R} \mathbf{d}_i \times \boldsymbol{\omega} \quad (25)$$

where, $\mathbf{R}\mathbf{d}_i = \mathbf{d}_i^0$, defined in Eq. 14, satisfies $\tilde{\mathbf{d}}_i^0 = \mathbf{R}\tilde{\mathbf{d}}_i\mathbf{R}^T$. Hence, the previous equation can be written as,

$$\dot{\mathbf{i}}_i = \dot{\mathbf{p}}_x - \mathbf{R}\tilde{\mathbf{d}}_i\mathbf{R}^T\boldsymbol{\omega} = \begin{bmatrix} \mathbf{I}_{3 \times 3} & -\mathbf{R}\tilde{\mathbf{d}}_i\mathbf{R}^T \end{bmatrix} \mathbf{v} \quad (26)$$

As seen in Section 2.2.1,

$$\dot{\mathbf{i}}_i = \begin{bmatrix} \mathbf{I}_{3 \times 3} & -\mathbf{R}\tilde{\mathbf{d}}_i\mathbf{R}^T \end{bmatrix} \mathbf{T}_R \dot{\mathbf{x}} \quad (27)$$

where, if roll-pitch-yaw angles are used, $\mathbf{T}_R$ can be calculated (Eq. 7),

$$\mathbf{T}_R = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{E} \end{bmatrix} \quad (28)$$

and,

$$\mathbf{E} = \mathbf{R} \begin{bmatrix} 1 & 0 & -\sin\beta \\ 0 & \cos\alpha & \cos\beta \sin\alpha \\ 0 & -\sin\alpha & \cos\beta \cos\alpha \end{bmatrix} \quad (29)$$

Combining Eqs. 27 and 28,

$$\dot{\mathbf{i}}_i = \underbrace{\begin{bmatrix} \mathbf{I}_{3 \times 3} & -\mathbf{R}\tilde{\mathbf{d}}_i\mathbf{R}^T \end{bmatrix} \mathbf{T}_R}_{\mathbf{J}_{x_i}} \dot{\mathbf{x}} = - \underbrace{\begin{bmatrix} -c\varphi_i c\theta_i & l_i c\varphi_i s\theta_i & l_i c\theta_i s\varphi_i \\ -c\varphi_i s\theta_i & -l_i c\varphi_i c\theta_i & l_i s\varphi_i s\theta_i \\ -s\varphi_i & 0 & -l_i c\varphi_i \end{bmatrix}}_{\mathbf{J}_{q_i}} \dot{\mathbf{q}}_i \quad (30)$$

Hence, the link velocity equation can be written in matrix form,

$$\mathbf{J}_{x_i} \dot{\mathbf{x}} = -\mathbf{J}_{q_i} \dot{\mathbf{q}}_i \quad (31)$$

where $\dot{\mathbf{q}}_i = [\dot{l}_i \quad \dot{\theta}_i \quad \dot{\varphi}_i]^T$.

Operating, the *link Jacobian matrix* associated with the serial chain i being $\mathbf{J}_i$ can be calculated,

$$-\mathbf{J}_{q_i} \dot{\mathbf{q}}_i = \mathbf{J}_{x_i} \dot{\mathbf{x}} \rightarrow \dot{\mathbf{q}}_i = - \underbrace{\mathbf{J}_{q_i}^{-1}}_{3 \times 3} \underbrace{\mathbf{J}_{x_i}}_{3 \times 6} \dot{\mathbf{x}} = \underbrace{\mathbf{J}_i}_{3 \times 6} \dot{\mathbf{x}} \quad (32)$$

All velocity relations can be deduced operating $\mathbf{J}_i$. Note that each row of $\mathbf{J}_i$ is associated to a joint variable contained in $\mathbf{q}_i$. Hence, if $[\mathbf{A}]_i$ is defined as the row operator that extracts the i -th row of a matrix $\mathbf{A}$,

$$\begin{aligned} \dot{\mathbf{q}}_a &= \mathbf{J}_{q_a} \dot{\mathbf{x}} & \mathbf{J}_{q_a} &= \begin{bmatrix} [\mathbf{J}_1]_1^T & \dots & [\mathbf{J}_6]_1^T \end{bmatrix}^T \\ \dot{\mathbf{q}}_{na} &= \mathbf{J}_{q_{na}} \dot{\mathbf{x}} & \mathbf{J}_{q_{na}} &= \begin{bmatrix} [\mathbf{J}_1]_2^T & \dots & [\mathbf{J}_6]_2^T & [\mathbf{J}_1]_3^T & \dots & [\mathbf{J}_6]_3^T \end{bmatrix}^T \end{aligned} \quad (33)$$

and combining both *active joint Jacobian* $\mathbf{J}_{\mathbf{q}_a}$ and *nonactuated joint Jacobian* $\mathbf{J}_{\mathbf{q}_{na}}$, the *joint variables Jacobian* $\mathbf{J}_{\mathbf{q}}$ can be calculated,

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{\mathbf{q}}_a \\ \dot{\mathbf{q}}_{na} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{\mathbf{q}_a} \\ \mathbf{J}_{\mathbf{q}_{na}} \end{bmatrix} \dot{\mathbf{x}} = \underbrace{\mathbf{J}_{\mathbf{q}}}_{18 \times 6} \dot{\mathbf{x}} \quad (34)$$

If only the control coordinates are considered, the *control coordinates Jacobian* $\mathbf{J}_{\mathbf{q}_c}$ can be calculated by combining the rows of $\mathbf{J}_{\mathbf{q}}$ that are related to sensorized (vector $\mathbf{I}_c$) joint variables,

$$\dot{\mathbf{q}}_c = [\mathbf{J}_{\mathbf{q}}]_{\mathbf{I}_c} \dot{\mathbf{x}} = \mathbf{J}_{\mathbf{q}_c} \dot{\mathbf{x}} \rightarrow \dot{\mathbf{x}} = \mathbf{J}_{\mathbf{q}_c}^\dagger \dot{\mathbf{q}}_c \quad (35)$$

where $\mathbf{J}_{\mathbf{q}_c}^\dagger$ is the Moore-Penrose generalized pseudoinverse of the Jacobian matrix associated with the control coordinates.

The rows not included in $\mathbf{J}_{\mathbf{q}_c}$, are associated to passive joint variables that are not sensorized nor actuated, hence, the *passive variable joint Jacobian* can be defined as,

$$\dot{\mathbf{q}}_p = \mathbf{J}_{\mathbf{q}_p} \dot{\mathbf{x}} \quad (36)$$

The relationship between the velocity of all joint coordinates $\dot{\mathbf{q}}$ and active ones $\dot{\mathbf{q}}_a$ is required to calculate the dynamic model of the robot, and can be calculated based on previously calculated Jacobians,

$$\dot{\mathbf{q}} = \begin{bmatrix} \mathbf{I}_3 \\ \mathbf{J}_{\mathbf{q}_{na}} \mathbf{J}_{\mathbf{q}_a}^{-1} \end{bmatrix}^T \dot{\mathbf{q}}_a = \mathbf{T} \dot{\mathbf{q}}_a \quad (37)$$

Finally, if control coordinates are to be considered, the relationship between velocity of all joint coordinates $\dot{\mathbf{q}}$ and control coordinates $\dot{\mathbf{q}}_c$,

$$\dot{\mathbf{q}} = \underbrace{\mathbf{T}_{\mathbf{q}}}_{18 \times n_c} \dot{\mathbf{q}}_c, \quad [\mathbf{T}_{\mathbf{q}}]_i = \begin{cases} \mathbf{e}_j^T, & q_i = q_{c_j} \in \mathbf{q}_c \\ [\mathbf{J}_{\mathbf{q}_p} \mathbf{J}_{\mathbf{p}}]_k, & q_i = q_{p_k} \in \mathbf{q}_p \end{cases} \quad (38)$$

where

$$[\mathbf{e}_j]_i = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$

Acceleration Problem. Acceleration relationships can be calculated by differentiating the link velocity equation Eq. 32 with respect to time,

$$\begin{aligned}\ddot{\mathbf{q}}_i &= \mathbf{J}_{q_i}^{-1} \mathbf{J}_{x_i} \ddot{\mathbf{x}} - \mathbf{J}_{q_i}^{-1} (\dot{\mathbf{J}}_{x_i} + \dot{\mathbf{J}}_{q_i} \mathbf{J}_i) \dot{\mathbf{x}} \\ \ddot{\mathbf{q}}_i &= \mathbf{J}_i \ddot{\mathbf{x}} + \dot{\mathbf{J}}_i \dot{\mathbf{x}}\end{aligned}\quad (39)$$

If the procedure defined for velocity relations is followed, the *acceleration Jacobian associated to the joint variables* $\dot{\mathbf{J}}_q$ can be calculated,

$$\begin{aligned}\ddot{\mathbf{q}} &= \begin{bmatrix} \mathbf{J}_{q_a} \\ \mathbf{J}_{q_{na}} \end{bmatrix} \ddot{\mathbf{x}} + \begin{bmatrix} \dot{\mathbf{J}}_{q_a} \\ \dot{\mathbf{J}}_{q_{na}} \end{bmatrix} \dot{\mathbf{x}} \\ \ddot{\mathbf{q}} &= \mathbf{J}_q \ddot{\mathbf{x}} + \dot{\mathbf{J}}_q \dot{\mathbf{x}}\end{aligned}\quad (40)$$

Similarly, if control coordinates are considered and the associated rows selected from the previous equation,

$$\ddot{\mathbf{q}}_c = \mathbf{J}_{q_c} \ddot{\mathbf{x}} + \dot{\mathbf{J}}_{q_c} \dot{\mathbf{x}} \rightarrow \ddot{\mathbf{x}} = \mathbf{J}_{q_c}^\dagger \ddot{\mathbf{q}}_c - \mathbf{J}_p \dot{\mathbf{J}}_{q_c} \mathbf{J}_p \dot{\mathbf{q}}_c = \mathbf{J}_p \ddot{\mathbf{q}}_c + \dot{\mathbf{J}}_p \dot{\mathbf{q}}_c \quad (41)$$

Finally, considering the time derivative of matrix $\mathbf{T}_q$,

$$\ddot{\mathbf{q}} = \underbrace{\mathbf{T}_q}_{18 \times n_c} \ddot{\mathbf{q}}_c + \underbrace{\dot{\mathbf{T}}_q}_{18 \times n_c} \dot{\mathbf{q}}_c \quad (42)$$

2.2.4. Dynamic Model

As stated previously, a subsystem approach will be used to calculate the dynamic model. Hence, the dynamic model of the moving platform and the limbs will be calculated independently in terms of their natural coordinates.

The Lagrangian formulation with multipliers will be used, as it provides the possibility to define the model in terms of any set of generalized coordinates. For that purpose, the Lagrangian of the robot L has to be calculated in terms of the kinetic K_{MP} , K_S and potential U_{MP} , U_S energies of each subsystem,

$$L = (K_{MP} - U_{MP}) + (K_S - U_S) \quad (43)$$

The calculation of these terms will be detailed for the Gough-Stewart platform next,

Kinetic and Potential Energies of the Moving Platform The moving platform is considered a free body rotating and translating in space with mass m_p and a centroidal inertia $\mathbf{I}_p$. Its center of mass (CM) is located in the origin of the moving reference frame $\mathbf{P}(u, v, w)$ (Fig. 1). As the TCP is located in the moving platform, the mass of the load m_c and its inertia $\mathbf{I}_c$ have to be considered in its dynamic model.

This way, if the energy of free motion in space is considered,

$$K_{MP} = \frac{1}{2} \dot{\mathbf{p}}_x^T \dot{\mathbf{p}}_x (m_p + m_c) + \frac{1}{2} \boldsymbol{\omega}^T (\mathbf{I}_p + \mathbf{I}_c) \boldsymbol{\omega} \quad (44)$$

Taking into account Eq.7,

$$\begin{aligned} K_{MP} &= \frac{1}{2} \dot{\mathbf{x}}^T \mathbf{T}_R^T \begin{bmatrix} (m_p + m_c) \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_c + \mathbf{I}_p \end{bmatrix} \mathbf{T}_R \dot{\mathbf{x}} \\ &= \frac{1}{2} \dot{\mathbf{x}}^T \underbrace{\mathbf{M}_x(\mathbf{x})}_{6 \times 6} \dot{\mathbf{x}} \end{aligned} \quad (45)$$

being $\mathbf{M}_x(\mathbf{x})$ the inertia matrix of the moving platform, defined in terms of $\mathbf{x}$.

The potential energy of the platform is related to its relative position with respect to the fixed frame,

$$U_{MP} = (m_p + m_c) g z \quad (46)$$

where g is gravity vector.

Kinetic and Potencial Energies of the Limbs Each of the limbs connecting the moving platform and the fixed base are structurally identical, being their parametric model identical. Each serial chain is a polar serial robot, presenting 3 dof associated to the link joint variables $\mathbf{q}_i = [l_i \ \theta_i \ \varphi_i]^T$ (Fig. 2). As seen in Fig. 4, the serial chain is composed by two elements: a cylinder of mass m_{i1} , attached using a universal joint to $\mathbf{A}_i$, and a rod, of mass m_{i2} , which has a prismatic motion with respect to the cylinder, and is attached to point $\mathbf{B}_i$.

The center of mass (CM) of each cylinder is calculated in terms of the non-actuated joint variables,

$$\mathbf{CM}_{i1} = \mathbf{a}_i + \begin{bmatrix} l_{ci1} \cos \theta_i \cos \varphi_i \\ l_{ci1} \sin \theta_i \cos \varphi_i \\ l_{ci1} \sin \varphi_i \end{bmatrix} \quad (47)$$

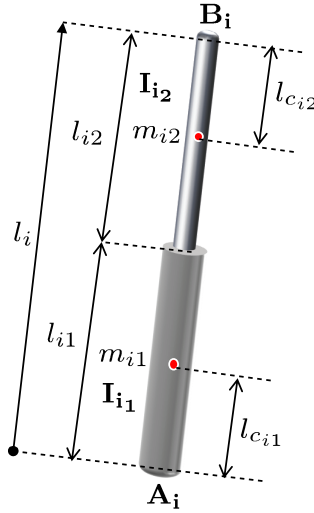


Figure 4. Limb parameters.

Differentiating this expression with respect to time, the linear velocity of the CM of each cylinder can be calculated,

$$\mathbf{v}_{i1} = \frac{\partial \mathbf{CM}_{i1}}{\partial \mathbf{q}} \dot{\mathbf{q}} = \mathbf{E}_{v_{i1}} \dot{\mathbf{q}} \quad (48)$$

where $\mathbf{E}_{v_{i1}}$ is the Jacobian that relates the linear velocity of the CM to the derivatives of the joint variables.

The angular velocity of the CM of the cylinder, ω_{i1} , can be calculated by projecting the velocity vectors associated to the nonactuated joint variables $\dot{\theta}_i$ and $\dot{\phi}_i$ in a moving frame attached to the CM of the cylinder,

$$\omega_{i1} = \begin{bmatrix} \dot{\theta}_i \sin \phi_i & -\dot{\phi}_i & \dot{\theta}_i \cos \phi_i \end{bmatrix}^T \quad (49)$$

If the terms are rearranged in terms of $\mathbf{q}$,

$$\omega_{i1} = \mathbf{E}_{\omega_{i1}} \dot{\mathbf{q}} \quad (50)$$

where $\mathbf{E}_{\omega_{i1}}$ is the Jacobian matrix that relates the angular velocity to the derivatives of the joint variables.

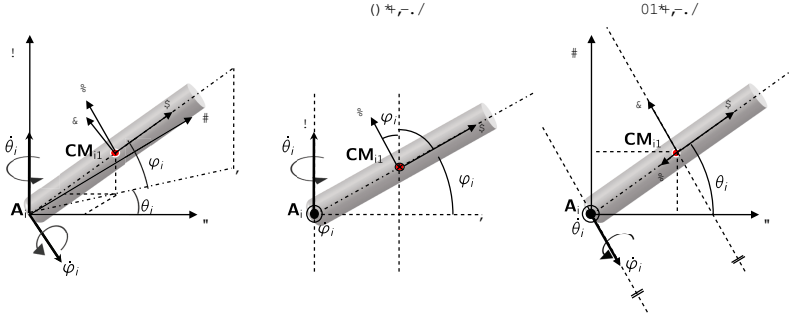


Figure 5. Cylinder rotation.

Once defined the velocities for the cylinder, the kinetic energy expression can be calculated,

$$K_{s i 1} = \frac{1}{2} \dot{\mathbf{q}}^T \left(m_{i 1} \mathbf{E}_{v i 1}^T \mathbf{E}_{v i 1} + \mathbf{E}_{\omega i 1}^T \mathbf{I}_{i 1} \mathbf{E}_{\omega i 1} \right) \dot{\mathbf{q}} = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}_{s i 1} \dot{\mathbf{q}} \quad (51)$$

where $\mathbf{M}_{s i 1}$ is the inertia matrix of the cylinder.

The kinetic energy of the rod can be calculated following the same procedure. First, the position of the CM is calculated in the fixed frame,

$$\mathbf{CM}_{i 2} = \mathbf{a}_i + \begin{bmatrix} (l_i - l_{c i 2}) \cos \theta_i \cos \varphi_i \\ (l_i - l_{c i 2}) \sin \theta_i \cos \varphi_i \\ (l_i - l_{c i 2}) \sin \varphi_i \end{bmatrix} \quad (52)$$

Differentiating this expression with respect to time, the linear velocity of the CM of each rod can be calculated,

$$\mathbf{v}_{i 2} = \frac{\partial \mathbf{CM}_{i 2}}{\partial \mathbf{q}} \dot{\mathbf{q}} = \mathbf{E}_{v i 2} \dot{\mathbf{q}} \quad (53)$$

where $\mathbf{E}_{v i 2}$ is the Jacobian that relates the linear velocity of the CM to the derivatives of the joint variables.

On the other hand, as the relative motion of the rod with respect to the cylinder is a pure translation, the angular velocity of the rod will be the same as the one of the cylinder,

$$\omega_{i 2} = \omega_{i 1} = \mathbf{E}_{\omega i 2} \dot{\mathbf{q}} \quad (54)$$

where $\mathbf{E}_{\omega i 2} = \mathbf{E}_{\omega i 1}$

Hence, the kinetic energy of the rod,

$$K_{s_{i2}} = \frac{1}{2} \dot{\mathbf{q}}^T (m_{i2} \mathbf{E}_{\mathbf{v}_{i2}}^T \mathbf{E}_{\mathbf{v}_{i2}} + \mathbf{E}_{\omega_{i2}}^T \mathbf{I}_{i2} \mathbf{E}_{\omega_{i2}}) \dot{\mathbf{q}} = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}_{s_{i2}} \dot{\mathbf{q}} \quad (55)$$

The potential energy of both the cylinder and the rod can be calculated considering the z coordinate of their CM position,

$$\begin{aligned} U_{s_{i1}} &= m_{i1} g l_{ci1} \sin \varphi_i \\ U_{s_{i2}} &= m_{i2} g (l_i - l_{ci2}) \sin \varphi_i \end{aligned} \quad (56)$$

Thus, the total kinetic and potential energies of this subsystem is calculated considering all limbs,

$$\begin{aligned} K_S &= \sum_{i=1}^6 K_{s_{i1}} + \sum_{i=1}^6 K_{s_{i2}} = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}_s \dot{\mathbf{q}} \\ U_S &= \sum_{i=1}^6 U_{s_{i1}} + \sum_{i=1}^6 U_{s_{i2}} = \sum_{i=1}^6 [m_{i2} (l_i - l_{ci2}) + m_{i1} l_{ci1}] g \sin \varphi_i \end{aligned} \quad (57)$$

where $\mathbf{M}_s = \sum_{i=1}^6 (\mathbf{M}_{s_{i1}} + \mathbf{M}_{s_{i2}})$ is the total inertia matrix of the serial chain subsystem.

2.3. Dynamic Model

Once defined all energies, the Lagrangian of the robot can be calculated,

$$L = (K_{MP} - U_{MP}) + (K_S - U_S) = L_S(\mathbf{q}, \dot{\mathbf{q}}) + L_{MP}(\mathbf{x}, \dot{\mathbf{x}}) \quad (58)$$

Note that the Lagrangian associated to the serial chains is defined only in terms of the joint variables, while the Lagrangian of the moving platform is defined in terms of the output coordinates. These coordinates sum a total of $n + n_q = 24$. However, only 6 are independent, as the robot provides only 6 dof. Hence, a set of 18 constraints have to be applied to solve the model. By using the Lagrangian Multiplier formulation, these constraints can be implicitly introduced by using a set of 18 multipliers.

This way, if the formulation is applied to each subsystem, two vectorial equations can be obtained,

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L_S}{\partial \dot{q}_j} \right) - \frac{\partial L_S}{\partial q_j} &= \sum_{i=1}^{n_q} \lambda_i \frac{\partial \Gamma_i(\mathbf{x}, \mathbf{q})}{\partial q_j} + \tau_{q_j}, \quad j = 1, \dots, 18 \\ \frac{d}{dt} \left(\frac{\partial L_{MP}}{\partial \dot{x}_k} \right) - \frac{\partial L_{MP}}{\partial x_k} &= \sum_{i=1}^{n_q} \lambda_i \frac{\partial \Gamma_i(\mathbf{x}, \mathbf{q}_i)}{\partial x_k}, \quad k = 1, \dots, 6 \end{aligned} \quad (59)$$

where τ_{q_j} are the virtual torques associated to each joint variable in $\mathbf{q}$, if all were actuated, and $\Gamma_i(\mathbf{x}, \mathbf{q})$ is the vectorial loop equation for each kinematic loop (Eq.12).

Operating on the left part of the equation system, and rearranging, the classical structured form of the dynamics of robots can be defined,

$$\begin{aligned} \mathbf{D}_x \ddot{\mathbf{x}} + \mathbf{C}_x \dot{\mathbf{x}} + \mathbf{G}_x &= \left[\frac{\partial \Gamma(\mathbf{x}, \mathbf{q})}{\partial \mathbf{x}} \right]^T \boldsymbol{\lambda} \\ \mathbf{D}_s \ddot{\mathbf{q}} + \mathbf{C}_s \dot{\mathbf{q}} + \mathbf{G}_s &= \left[\frac{\partial \Gamma(\mathbf{x}, \mathbf{q})}{\partial \mathbf{q}} \right]^T \boldsymbol{\lambda} + \boldsymbol{\tau}_r \end{aligned} \quad (60)$$

where $\mathbf{D}_x$, $\mathbf{C}_x$ and $\mathbf{G}_x$ are the inertia, Coriolis and gravity terms associated with the mobile platform, and defined in terms of $\mathbf{x}$, $\mathbf{D}_s$, $\mathbf{C}_s$ and $\mathbf{G}_s$ are the inertia, Coriolis and gravity terms associated with the legs, defined in terms of $\mathbf{q}$, $\boldsymbol{\lambda}$ is the vector of the Lagrange multipliers and $[\tau_{r_j}]_{j=1, \dots, 18} = \boldsymbol{\tau}_r$.

Combining both equations, Lagrange multipliers can be eliminated,

$$\boldsymbol{\tau}_r = \mathbf{D}_s \ddot{\mathbf{q}} + \mathbf{C}_s \dot{\mathbf{q}} + \mathbf{G}_s - \underbrace{\left(\frac{\partial \Gamma}{\partial \mathbf{q}} \right)^T}_{18 \times 18} \underbrace{\left[\left(\frac{\partial \Gamma}{\partial \mathbf{x}} \right)^T \right]^\dagger}_{18 \times 6} (\mathbf{D}_x \ddot{\mathbf{x}} + \mathbf{C}_x \dot{\mathbf{x}} + \mathbf{G}_x) \quad (61)$$

and considering that $\Gamma(\mathbf{x}, \mathbf{q}) = \mathbf{0}_{18 \times 1}$ is the set of all constraints of the robot (Eq. 16), the constraint Jacobian $\mathbf{J}_C = \partial \mathbf{x} / \partial \mathbf{q}$ can be defined,

$$\boldsymbol{\tau}_r = \mathbf{D}_s \ddot{\mathbf{q}} + \mathbf{J}_C^T \mathbf{D}_x \ddot{\mathbf{x}} + \mathbf{C}_s \dot{\mathbf{q}} + \mathbf{J}_C^T \mathbf{C}_x \dot{\mathbf{x}} + \mathbf{G}_s + \mathbf{J}_C^T \mathbf{G}_x \quad (62)$$

However, as previously stated, $\boldsymbol{\tau}_r$ is the set of virtual torques associated with all joint variables in $\mathbf{q}$. In parallel robots, only the actuated set $\mathbf{q}_a$ has an actuator attached and can provide torque, hence, the model needs to be projected to the actuated space using the Jacobian defined in Eq. 37 [29],

$$\boldsymbol{\tau}_{q_a} = \mathbf{T}^T \boldsymbol{\tau}_r = \mathbf{T}^T \mathbf{D}_s \ddot{\mathbf{q}} + \mathbf{T}^T \mathbf{J}_C^T \mathbf{D}_x \ddot{\mathbf{x}} + \mathbf{T}^T \mathbf{C}_s \dot{\mathbf{q}} + \mathbf{T}^T \mathbf{J}_C^T \mathbf{C}_x \dot{\mathbf{x}} + \mathbf{T}^T \mathbf{G}_s + \mathbf{T}^T \mathbf{J}_C^T \mathbf{G}_x \quad (63)$$

where it can be easily demonstrated that $\mathbf{T}^T \mathbf{J}_C^T = [\mathbf{J}_{q_a}^{-1}]^T$,

$$\boldsymbol{\tau}_{q_a} = \mathbf{T}^T \mathbf{D}_s \ddot{\mathbf{q}} + [\mathbf{J}_{q_a}^{-1}]^T \mathbf{D}_x \ddot{\mathbf{x}} + \mathbf{T}^T \mathbf{C}_s \dot{\mathbf{q}} + [\mathbf{J}_{q_a}^{-1}]^T \mathbf{C}_x \dot{\mathbf{x}} + \mathbf{T}^T \mathbf{G}_s + [\mathbf{J}_{q_a}^{-1}]^T \mathbf{G}_x \quad (64)$$

The previous equation defines the dynamic model of the robot in terms of all variables. If the redundant data is to be used to feed the dynamic model, the

model can be rewritten in terms of $\mathbf{q}_c$ in compact form using the velocity and acceleration expressions defined previously,

$$\boldsymbol{\tau}_{q_a} = \mathbf{D}\ddot{\mathbf{q}}_c + \mathbf{C}\dot{\mathbf{q}}_c + \mathbf{G} \quad (65)$$

where

$$\begin{aligned} \mathbf{D} &= \mathbf{T}^T \mathbf{D}_q \mathbf{T}_q + [\mathbf{J}_{q_a}^{-1}]^T \mathbf{D}_x \mathbf{J}_p \\ \mathbf{C} &= \mathbf{T}^T (\mathbf{D}_q \dot{\mathbf{T}}_q + \mathbf{C}_q \mathbf{T}_q) + [\mathbf{J}_{q_a}^{-1}]^T (\mathbf{D}_x \dot{\mathbf{J}}_p + \mathbf{C}_x \mathbf{J}_p) \\ \mathbf{G} &= \mathbf{T}^T \mathbf{G}_q + [\mathbf{J}_{q_a}^{-1}]^T \mathbf{G}_x \end{aligned}$$

which defines the dynamic model of the robot in terms of any set of control variables $\mathbf{q}_c$, which can include any set of sensorized variables. Note that the matrices of the model are not regular, as the model is fed with more information than strictly needed. This will be the basis of the Extended control approaches analyzed in further sections.

2.4. Model Validation

In order to validate the approach, a particular case will be analyzed. The works of Wang and Gosselin [43], Tsai [41] and Guo and Li [14] have established a well-documented study case that provides all the required data for proper model validation. Hence, the redundant model results will be compared with the ones of these works in order to validate the approach.

The particular Gough-Stewart platform parameters used are shown in Table 1, and have been extracted from the aforementioned works.

The reference trajectories proposed by these authors are detailed next. The first one is a constant orientation trajectory, where $\alpha = \beta = \gamma = 0$ and

$$\mathbf{p}_x = \begin{bmatrix} -1.5 + 0.2 \sin \omega t \\ 0.2 \sin \omega t \\ 1 + 0.2 \sin \omega t \end{bmatrix}$$

where $\omega = 3 \text{ rad/s}$ and $\omega t = [0, 2\pi]$.

The second trajectory is a constant position trajectory $\mathbf{p}_x = [-1.5 \ 0 \ 1]^T$, where platform rotates with respect to the fixed z axis: $\alpha = \beta = 0$ and $\gamma = 0.35 \sin \omega t$ with $\omega = 3 \text{ rad/s}$ and $\omega t = [0, 2\pi]$.

Using the dynamic model detailed in Eq. 65 the torques $\boldsymbol{\tau}_{q_a}$ required to execute these trajectories have been calculated for the case in which all joint variables are sensorized, this is, $\mathbf{q}_c = \mathbf{q}$. The obtained results can be seen in Figs.

Table 1. Dynamic and kinematic parameters ($i = 1, \dots, 6$ limbs)

Parameter	value
$\mathbf{a}_1$	$\begin{bmatrix} -2.12 & 1.374 & 0 \end{bmatrix}^T$ (m)
$\mathbf{a}_2$	$\begin{bmatrix} -2.38 & 1.224 & 0 \end{bmatrix}^T$ (m)
$\mathbf{a}_3$	$\begin{bmatrix} -2.38 & -1.224 & 0 \end{bmatrix}^T$ (m)
$\mathbf{a}_4$	$\begin{bmatrix} -2.12 & -1.374 & 0 \end{bmatrix}^T$ (m)
$\mathbf{a}_5$	$\begin{bmatrix} 0 & -0.15 & 0 \end{bmatrix}^T$ (m)
$\mathbf{a}_6$	$\begin{bmatrix} 0 & 0.15 & 0 \end{bmatrix}^T$ (m)
$\mathbf{d}_1$	$\begin{bmatrix} 0.17 & 0.595 & -0.4 \end{bmatrix}^T$ (m)
$\mathbf{d}_2$	$\begin{bmatrix} -0.6 & 0.15 & -0.4 \end{bmatrix}^T$ (m)
$\mathbf{d}_3$	$\begin{bmatrix} -0.6 & -0.15 & -0.4 \end{bmatrix}^T$ (m)
$\mathbf{d}_4$	$\begin{bmatrix} 0.17 & -0.595 & -0.4 \end{bmatrix}^T$ (m)
$\mathbf{d}_5$	$\begin{bmatrix} 0.43 & -0.445 & -0.4 \end{bmatrix}^T$ (m)
$\mathbf{d}_6$	$\begin{bmatrix} 0.43 & 0.445 & -0.4 \end{bmatrix}^T$ (m)
l_{i1}	0.5 (m)
$lc_{l_{i1}}$	0.25 (m)
l_{i2}	0.5 (m)
$lc_{l_{i2}}$	0.25 (m)
m_p	1.5 (kg)
m_c	0 (kg)
m_{i1}	0.1 (kg)
m_{i2}	0.1 (kg)
$\mathbf{I}_p$	$0.08 \mathbf{I}_3$ (kg m ²)
$\mathbf{I}_{i1}$	$0.00625 \mathbf{I}_3$ (kg m ²)
$\mathbf{I}_{i2}$	$0.00625 \mathbf{I}_3$ (kg m ²)
$\mathbf{I}_c$	$\mathbf{0}_3$ (kg m ²)

6-7, and are identical to those obtained by Wang and Gosselin [43], Tsai [41] and Guo and Li [14]. Hence, for the same reference trajectories and model parameters, the redundant dynamic model yields the same results as those obtained by the cited authors. Moreover, any set of $\mathbf{q}_c$ that includes at least the actuated variables $\mathbf{q}_a$ yields the same result.

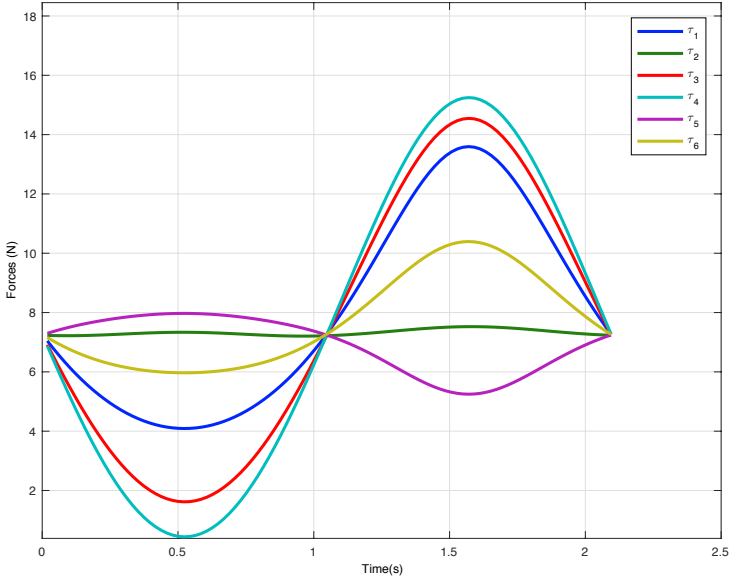


Figure 6. Forces for the constant orientation trajectory.

3. Control

3.1. Position Control in Parallel Robots

In the control area of parallel robots, the main trend has been to import directly the control approaches from serial robots to parallel robots without considering the specific features of the latter ones. This way, most of the position control related works in the literature fall into two categories: local feedback control approaches and model based control approaches.

Local feedback control approaches are based on the use of a local PID based controller to control each joint independently from the rest of the mechanism. In this group several approaches can be found: simple Proportional-Integral-Derivative (PID) loop based schemes [7, 16, 36], joint dynamic model based approaches [6, 13], Proportional-Derivative (PD) plus gravitation compensations schemes [12, 39] or nonlinear PID based control laws [30, 38] fall into this group. In general, these approaches are easy to implement due to their simple control law. However, as they do not consider the dynamic coupling between the joints, they do not provide good control performance when high speed and accuracy

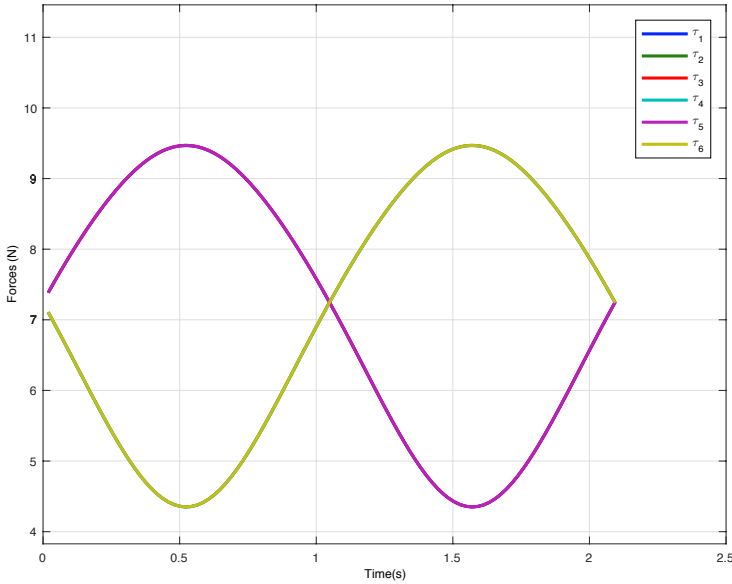


Figure 7. Forces for the constant position trajectory.

are required.

Model based approaches, on the other hand, require the dynamic model of the parallel robot to define the control law and provide better performance when high speed, accelerations and accuracy are needed. One of the most extended approach is the Computed Torque Control (CTC) scheme [1, 8, 11]. However, obtaining an accurate dynamic model is a challenging task, as simplifications are usually made and some phenomena, such as friction or backlash, are not considered. Thus, in order to ensure the dynamic performance of the controller in presence of model parameter uncertainties, some authors have also proposed robust [13, 17, 19, 21] or adaptive control [15, 33] approaches.

From the results offered by the works in the literature, model-based schemes seem to be the best approach when high-speed and accuracy requirements must be met in parallel robots. However, the correct calibration of the dynamic model is difficult, as parameter identification errors arise, and their computational cost is higher than the simpler, local feedback controllers.

3.2. The Extended CTC Approach

An interesting approach to increase control performance in parallel robots is the introduction of extra sensors in the nonactuated joints of the mechanism, so that the redundant data can be used to minimize the effect of parameter uncertainties and to implement a more efficient redundant control law in parallel robots. Although this approach increases the cost of the robot, the advantages of using redundant sensor data have been demonstrated by authors such as Merlet [25], Baron and Angeles [3], Marquet *et al* [24], and Bauma *et al* [4].

The Extended CTC approach is based on this idea [45]. This position control scheme is based on a modified Computed Torque Control approach that makes use of the redundant dynamic model defined in Section 2.2. This way, the Extended CTC approach combines the performance of the classical CTC with the robustness of sensor redundancy, providing better trajectory tracking than the classical nonredundant CTC approach.

The control law is defined as (Fig. 8),

$$\tau_{qa} = \hat{\mathbf{D}}(\mathbf{q}_c, \mathbf{q}_p, \mathbf{x}) (\ddot{\mathbf{q}}_{cd} + \mathbf{K}_v \dot{\mathbf{e}}_{qc} + \mathbf{K}_p \mathbf{e}_{qc}) + \hat{\mathbf{C}}(\mathbf{q}_c, \mathbf{q}_p, \mathbf{x}, \dot{\mathbf{q}}_c, \dot{\mathbf{q}}_p, \dot{\mathbf{x}}) \dot{\mathbf{q}}_c + \hat{\mathbf{G}}(\mathbf{q}_c, \mathbf{q}_p, \mathbf{x}) \quad (66)$$

where $\mathbf{K}_p$ and $\mathbf{K}_v$ are the position and velocity matrix gains, $\mathbf{e}_{qc} = \mathbf{q}_{cd} - \mathbf{q}_c$ is the tracking error of the control coordinates and $\dot{\mathbf{e}}_{qc}$ its time derivative, $\mathbf{q}_{cd}$ is the reference of the control coordinates, and $\hat{\mathbf{D}}$, $\hat{\mathbf{C}}$, $\hat{\mathbf{G}}$ are the identified matrices of the dynamic model defined in Eq. 65

This way, all the measurable variables contained in $\mathbf{q}_c$ are used to feed the model, and in case of errors in model parameters, the controller tries to minimize not only the errors in active joints, but also the sensorized ones, achieving better dynamic performance.

3.3. Extended CTC Control Validation

The Gough-Stewart platform defined in Section 2.2 is used to validate the performance of the Extended CTC approach over the classical CTC scheme. For that purpose, the redundant dynamic model defined in Eq. 65 is calculated for the four different sensor configurations detailed in Table 2. As it can be seen, Configuration 1 corresponds to the classical CTC approach in which only the active joints are sensorized, while Configuration 4 corresponds to the case in which all joint variables are measured.

A complex, high speed helicoidal trajectory defined in the non-singular workspace of the Gough Platform (Fig. 9) is defined to compare the dynamic

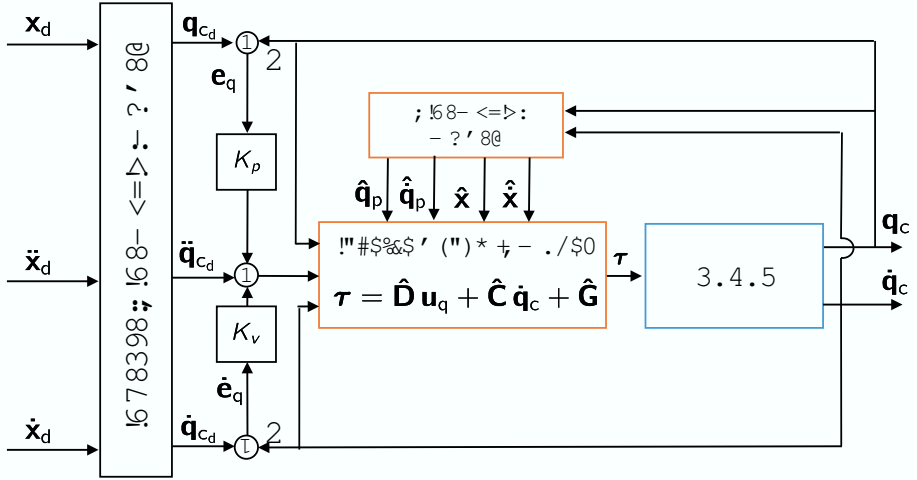


Figure 8. Extended Computed Torque Control.

Table 2. Extended CTC Configurations

Config.	$\mathbf{q}_c$
1	$\mathbf{q}_c = \mathbf{q}_a \in \mathbb{R}^6$
2	$\mathbf{q}_c = [\mathbf{q}_a^T \quad \theta_1 \quad \theta_3 \quad \varphi_1 \quad \varphi_3]^T \in \mathbb{R}^{10}$
3	$\mathbf{q}_c = [\mathbf{q}_a^T \quad \varphi^T]^T \in \mathbb{R}^{12}$
4	$\mathbf{q}_c = \mathbf{q} \in \mathbb{R}^{18}$

performance of the approach. This trajectory, defined in Eq. 67, allows the controller to be tested in a wide workspace area.

$$\mathbf{x}_{\text{ref}} = \begin{bmatrix} 0.4 \cos(10t) \\ 0.4 \sin(10t) \\ 1.2 + 0.1592t \\ 10\pi/180 \sin(12t) \\ 10\pi/180 \cos(12t) \\ 10\pi/180 \sin(12t) \end{bmatrix} \quad (67)$$

The model parameter set for the comparative analysis is the one detailed in

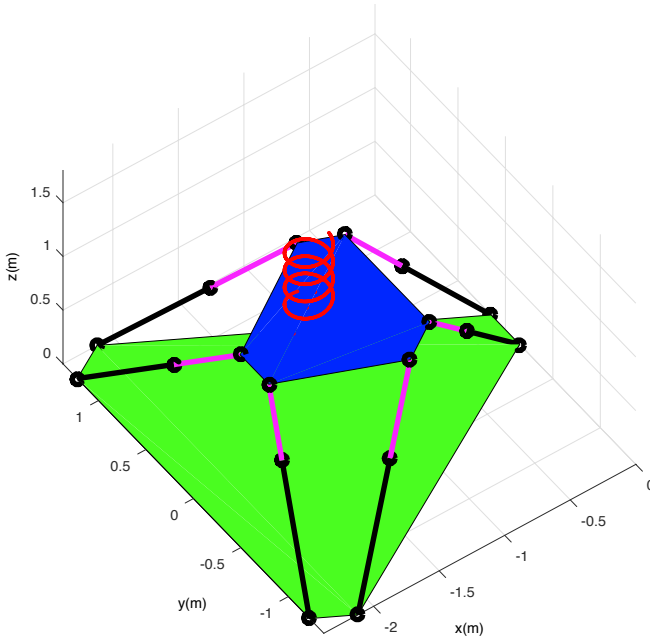


Figure 9. Helicoidal Reference Trajectory.

Table 1 where, in addition, a load of mass $m_c = 1$ kg and inertia $\mathbf{I}_c = 0.06\mathbf{I}_3$ kg m^2 has been attached to the platform. Moreover, a friction term $\mathbf{F}_r$ is introduced into the direct dynamic model to simulate the dynamics of the robot, which is not considered in the control law,

$$\mathbf{F}_r = \mathbf{T}^T \mathbf{F}_v \dot{\mathbf{q}} + \mathbf{T}^T \mathbf{F}_c \text{sign}(\dot{\mathbf{q}}) \quad (68)$$

where $\mathbf{F}_v$ is the viscous friction term and $\mathbf{F}_c$ is the Coulomb term. The $\mathbf{F}_v$ terms associated with the active joints $\mathbf{q}_a$ are assigned a value of 0.83 kg m/s , and those associated with the non-actuated joints are assigned a value of 0.09 kg m/s . Further, the terms of $\mathbf{F}_c$ associated with the active joints $\mathbf{q}_a$ are 0.42 N , while those associated with the non-actuated joints are 0.05 N .

The proportional K_p and velocity K_v gains have been tuned to obtain a maximum overshoot of 5% and a peak time of 0.05 s in the classical CTC controller. The same gains are used for all controllers, and the cycle time is set to 5 ms.

In order to carry out the comparative analysis, the Integral of the Absolute Error (IAE) performance index of the end-effector positioning and orientation

error, defined as $\int_0^\infty |e(t)| dt$, will be analyzed. Specifically, for the proposed reference trajectories the real end-effector coordinates $\mathbf{x}$ are measured and compared with the desired ones in order to calculate the tracking accuracy. In order to show the effect of parameter uncertainty in the performance of the controllers, the real robot parameters have been randomly modified from their nominal values by a maximum value of 0.1 mm in distance parameters, 0.01 kg in mass parameters and up to the 10% in inertia terms. Five parameter cases have been considered, from 0% to 100% of these maximum values. In order to obtain statistically relevant data, each simulation iteration and case has been repeated 350 times using different parameter variations within the desired bounds and the average dynamic performance is considered as representative for a given controller.

Results are summarized in Figs. 10 and 11. As it can be seen, the IAE error index increases with the parameter variation percentage, this is, errors increase as uncertainties arise. However, the relative increase is much lower when redundant data is used (Configurations 2, 3 and 4).

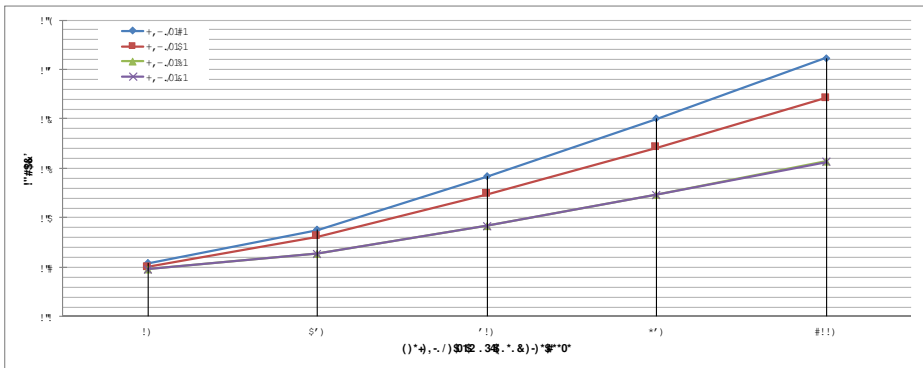


Figure 10. Average positioning IAE.

The relative performance also varies if different sensor configurations are considered. For instance, it seems logical to think that the more sensors used, the better the performance will be. However, this is not always straightforward, as Configuration 2 and 3 have very similar performance with different sensor requirements. Thus, it is clear that it is necessary to identify an appropriate sensor distribution in order to achieve the best performance and exploit the potential of the Extended CTC approach.

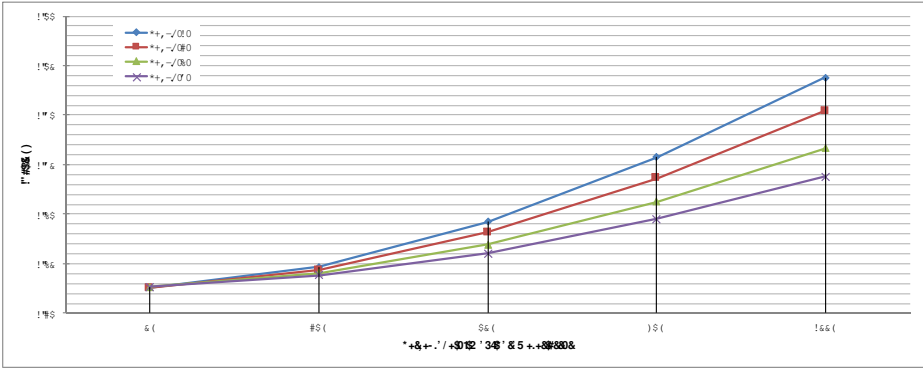


Figure 11. Average orientation IAE.

As a more detailed example, is shown in Figs. 12 and 13 , where the error over time associated with the two TCP locations is shown for a case in which a helicoidal trajectory is followed and maximum errors are introduced in the nominal parameters. Configuration 1 and 4 are compared, being Configuration 1 the equivalent of the classical CTC approach and Configuration 4 the one with all joint variables sensorized. If errors in the joint variables are considered, it can be seen that the classical approach provides better tracking in the active joints, as the controller tries to minimize these errors, while the Extended CTC seems to present higher errors in the sensorized joints, as errors are distributed among the measured variables. However, this error distribution provides better results in the task space, where Configuration 4 presents a smaller trajectory tracking.

Thus, in summary, it has been demonstrated that the use of the redundant Extended CTC configurations can provide better performance than the classical CTC approaches even in the presence of model parameter errors and nonlinear phenomena such as friction. However, it is clear that in order to achieve the best results the sensor configuration has to be selected with care.

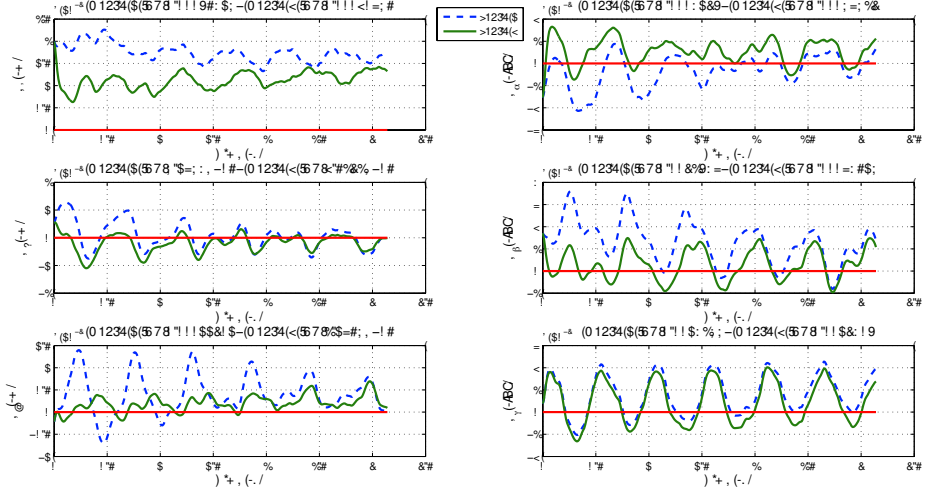


Figure 12. Time evolution of the TCP position and orientation error.

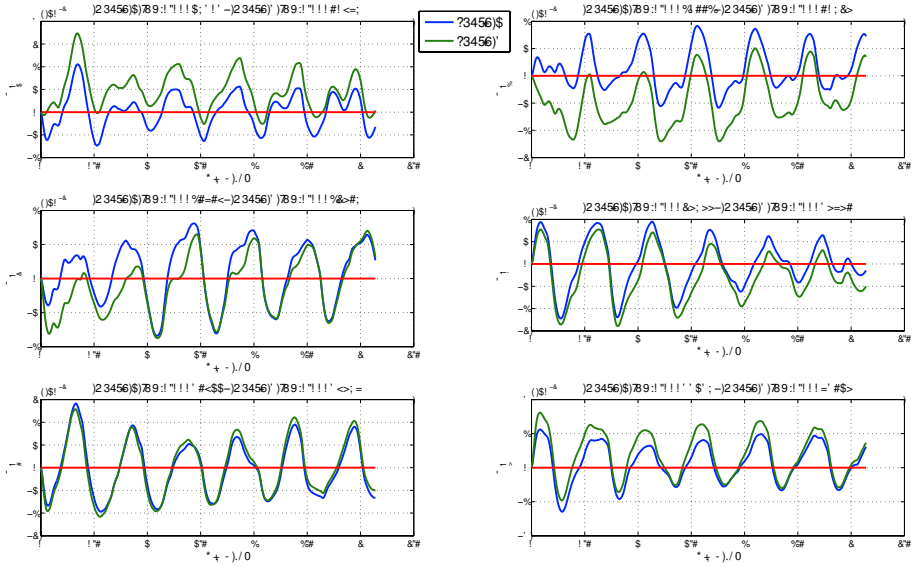


Figure 13. Time evolution of joint variables error.

4. Sensitivity

4.1. Sensitivity-Based Approach to Evaluate Sensor Configurations

One of the central issues when implementing redundant sensor based approaches such as the Extended CTC is determining the number and location of the extra sensors. This is an important step in the design of the mechanism, as the introduction of extra sensors is not a trivial task. Hence, determining how many sensors are needed and where are they located is very useful when designing passive joints.

Note that the Extended CTC is defined in a general way in terms of $\mathbf{q}_c$. Thus, multiple sensor configurations can be used to implement it. Depending on the complexity of the mechanism, the number of possible combinations can increase significantly.

In order to evaluate in an easy way the different sensor configurations for a given platform and task, a sensitivity based procedure is proposed. This analysis allows to determine the influence of the variation of each model parameter in the position of the TCP. Thus, if the sensitivity of all parameters for a given Extended CTC configuration is calculated, a measure of the relative robustness to model parameter uncertainties can be obtained. Hence, if different Extended CTC configurations are analyzed, an insight into the relative robustness of each configuration can be calculated, so that the most suitable sensor distribution can be selected for a given robot and trajectory.

The measure of the sensitivity of a sensor configuration is based on the procedure proposed in [42] and modified to suit the needs of parallel robots. This procedure is based on the calculation of the sensitivity function $\xi_{\theta_k}^x$ that calculates the deviation of the TCP $\delta\mathbf{x}$ for the Extended CTC approach along a nominal trajectory P_d when the parameter θ_k is uncertain. This is calculated as the difference between the deviation of the task coordinates in the nominal case, $\delta\mathbf{x}$, and the deviation of the task coordinates when the parameter θ_k varies from its nominal (or ideal) value, $\delta\mathbf{x}_{\theta_k}$,

$$\xi_{\theta_k}^x = \delta\mathbf{x} - \delta\mathbf{x}_{\theta_k} \quad (69)$$

where the calculation of both $\delta\mathbf{x}$ and $\delta\mathbf{x}_{\theta_k}$ for a given trajectory and Extended CTC configuration requires the linearization of the closed loop dynamics along the nominal trajectory P_d .

Hence, if the dynamics of the robot (Eq. 65) and the Extended CTC control law (Eq. 66) are considered,

$$\mathbf{D}\ddot{\mathbf{q}}_c + \mathbf{h} = \hat{\mathbf{D}}(\ddot{\mathbf{q}}_{cd} + \mathbf{K}_v \dot{\mathbf{e}}_q + \mathbf{K}_p \mathbf{e}_q) + \hat{\mathbf{h}} \quad (70)$$

where $\hat{\mathbf{D}}(\hat{\mathbf{x}}, \hat{\mathbf{q}}_p, \mathbf{q}_c)$ is the estimated inertia matrix and $\hat{\mathbf{h}} = \hat{\mathbf{C}}(\hat{\mathbf{x}}, \hat{\mathbf{q}}_p, \mathbf{q}_c, \hat{\mathbf{x}}, \hat{\mathbf{q}}_p, \dot{\mathbf{q}}_c) \dot{\mathbf{q}}_c + \hat{\mathbf{G}}(\hat{\mathbf{x}}, \hat{\mathbf{q}}_p, \mathbf{q}_c)$ groups the estimated Coriolis and gravity terms of the dynamic model implemented in the Extended CTC controller. As these matrices are estimated, their calculation will be made in terms of *uncertain* parameters, whose value will vary from the nominal value. On the other hand, the inertia matrix $\mathbf{D}$ and the nonlinear vector $\mathbf{h}$ characterize the *real* dynamics of the robot, whose calculation is done using the nominal parameters.

In order to calculate the linearized dynamics of the closed-loop system, first the left term of Eq. 70 will be linearized considering a nominal trajectory P_d ,

$$\begin{aligned} \delta\tau = & \mathbf{D}(\mathbf{x}, \mathbf{q}_p, \mathbf{q}_{cd}) \delta\ddot{\mathbf{q}}_c + \left. \frac{\partial \mathbf{D} \ddot{\mathbf{q}}_c}{\partial \mathbf{x}} \right|_{P_d} \delta \mathbf{x} + \left. \frac{\partial \mathbf{D} \ddot{\mathbf{q}}_c}{\partial \mathbf{q}} \right|_{P_d} \delta \mathbf{q} + \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right|_{P_d} \delta \mathbf{x} \\ & + \left. \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \right|_{P_d} \delta \mathbf{q} + \left. \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{x}}} \right|_{P_d} \delta \dot{\mathbf{x}} + \left. \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{q}}} \right|_{P_d} \delta \dot{\mathbf{q}} \end{aligned} \quad (71)$$

If the linearized model is to be defined in terms of the control coordinates $\mathbf{q}_c$, the velocity Jacobians detailed in Section 2.2.3 can be used to project the model,

$$\delta\tau = \mathbf{A}_m \delta\ddot{\mathbf{q}}_c + \mathbf{B}_m \delta\dot{\mathbf{q}}_c + \mathbf{C}_m \delta\mathbf{q}_c \quad (72)$$

where,

$$\begin{aligned} \mathbf{A}_m &= \mathbf{D}|_{P_d} \\ \mathbf{B}_m &= \left. \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{x}}} \right|_{P_d} \mathbf{J}_p + \left. \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{q}}} \right|_{P_d} \mathbf{T}_q \\ \mathbf{C}_m &= \left. \frac{\partial \mathbf{D} \ddot{\mathbf{q}}_c}{\partial \mathbf{x}} \right|_{P_d} \mathbf{J}_p + \left. \frac{\partial \mathbf{D} \ddot{\mathbf{q}}_c}{\partial \mathbf{q}} \right|_{P_d} \mathbf{T}_q \\ &\quad + \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right|_{P_d} \mathbf{J}_p + \left. \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \right|_{P_d} \mathbf{T}_q \\ &\quad + \left. \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{x}}} \right|_{P_d} \mathbf{J}_p + \left. \frac{\partial \mathbf{h}}{\partial \dot{\mathbf{q}}} \right|_{P_d} \dot{\mathbf{T}}_q \end{aligned}$$

where Eq. 72 is defined in terms of the nominal parameters of the model, i.e., the real ones.

If the same procedure is applied to the right term of Eq. 70,

$$\delta\tau = \hat{\mathbf{A}}_c \delta\mathbf{u} + \hat{\mathbf{B}}_c \delta\dot{\mathbf{q}}_c + \hat{\mathbf{C}}_c \delta\mathbf{q}_c \quad (73)$$

where $\delta\mathbf{u} = -\mathbf{K}_v \delta\dot{\mathbf{q}}_c - \mathbf{K}_p \delta\mathbf{q}_c$ [42]. Note that Eq. 73 is defined in terms of the *uncertain* parameters of the model.

Combining the linearized Eqs. 71 and 73, the linearized closed loop dynamics can be calculated as,

$$\mathbf{0} = \mathbf{A} \delta\ddot{\mathbf{q}}_c + \mathbf{B} \delta\dot{\mathbf{q}}_c + \mathbf{C} \delta\mathbf{q}_c \quad (74)$$

where $\mathbf{A} = \mathbf{A}_m$, $\mathbf{B} = \mathbf{B}_m - \hat{\mathbf{B}}_c + \hat{\mathbf{A}}_c \mathbf{K}_v$ and $\mathbf{C} = \mathbf{C}_m - \hat{\mathbf{C}}_c + \hat{\mathbf{A}}_c \mathbf{K}_p$.

In order to calculate the deviation of the nominal TCP $\delta\mathbf{x}$ pose required to analyze sensitivity, the kinematic relations of Section 2.2.3 can be used,

$$\delta\mathbf{q}_c = \mathbf{J}_p^\dagger \delta\mathbf{x} = \mathbf{J}_{qc} \delta\mathbf{x} \quad (75)$$

Hence, Eq. 74 can be rewritten as,

$$\delta\ddot{\mathbf{x}} = (-\mathbf{A} \mathbf{J}_{qc})^{-1} \left[(2\mathbf{A} \mathbf{J}_{qc} + \mathbf{B} \mathbf{J}_{qc}) \delta\dot{\mathbf{x}} + (\mathbf{A} \ddot{\mathbf{J}}_{qc} + \mathbf{B} \dot{\mathbf{J}}_{qc} + \mathbf{C} \mathbf{J}_{qc}) \delta\mathbf{x} \right] \quad (76)$$

that defines an ODE system, that can be solved for an initial $\delta\mathbf{x}_0$ and an arbitrary nominal reference trajectory $P_d(t)$. If the nominal parameters are used, Eq. 76 will be used to calculate $\delta\mathbf{x}$, while if a parameter θ_k is modified, i.e. is *uncertain*, the modified parameter will be used to calculate $\hat{\mathbf{A}}_c$, $\hat{\mathbf{B}}_c$ and $\hat{\mathbf{C}}_c$, and this expression will be used to calculate $\delta\mathbf{x}_{\theta_k}$. Then, applying Eq. 69, the sensitivity function over the reference trajectory will be obtained.

Note that the absolute value of the sensitivity function $\xi_{\theta_k}^x$ will depend on the selection of the initial $\delta\mathbf{x}_0$ used to integrate it. Thus, the analysis of the normalized sensitivity function is proposed $\xi_{\theta_k}^x / \delta\mathbf{x}_0$. This way, the characteristic sensitivity value will be considered as the maximum of the absolute value of the normalized sensitivity function.

4.2. Sensitivity Based Extended CTC Sensor Configuration Selection

In order to validate the sensitivity based Extended CTC sensor configuration selection methodology, the Gough-Stewart platform analyzed in Section 2.2 will be used as a study case.

In Table 3, the ten different sensor configurations to be evaluated are shown, being Config. 1 the traditional CTC approach with no extra sensors, and Config. 10 the one in which all joints are sensorized. In order to compare the sensitivity results of these configurations, the controller matrices $\mathbf{K}_p$ and $\mathbf{K}_v$ have been tuned to obtain an overshoot of 5% and a peak time of 0.1s for all configurations. The nominal values of the robot are those detailed in Table 1, considering a maximum identification error of 1mm for lengths, 10g for masses and a 10% variation of the inertia nominal value.

Table 3. Extended CTC configurations to be evaluated

Config 1	$\mathbf{q}_c = \mathbf{q}_a$
Config 2	$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a^T & \boldsymbol{\theta}^T \end{bmatrix}^T$
Config 3	$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a^T & \boldsymbol{\varphi}^T \end{bmatrix}^T$
Config 4	$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a^T & [\boldsymbol{\theta}]_{1,3,6}^T & [\boldsymbol{\varphi}]_{1,3,6}^T \end{bmatrix}^T$
Config 5	$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a^T & \theta_1 & \theta_3 & \varphi_1 & \varphi_3 \end{bmatrix}^T$
Config 6	$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a^T & \theta_1 & \theta_3 & \varphi_1 \end{bmatrix}^T$
Config 7	$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a^T & \theta_1 & \theta_3 & \varphi_3 \end{bmatrix}^T$
Config 8	$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a^T & \theta_1 & \theta_3 & \theta_6 \end{bmatrix}^T$
Config 9	$\mathbf{q}_c = \begin{bmatrix} \mathbf{q}_a^T & \varphi_1 & \varphi_3 & \varphi_6 \end{bmatrix}^T$
Config 10	$\mathbf{q}_c = \mathbf{q}$

As the proposed methodology requires a nominal trajectory to be evaluated, the helicoidal trajectory detailed in Eq. 67 (Fig. 9) will be used.

The sensitivity related to orientation and position coordinates will be evaluated separately. For that purpose, the following expressions will be used,

$$\begin{aligned} \xi_{\theta_k}^{POS} &= \sqrt{(\xi_{\theta_k}^x)^2 + (\xi_{\theta_k}^y)^2 + (\xi_{\theta_k}^z)^2} \\ \xi_{\theta_k}^{ORI} &= \sqrt{(\xi_{\theta_k}^\alpha)^2 + (\xi_{\theta_k}^\beta)^2 + (\xi_{\theta_k}^\gamma)^2} \end{aligned} \quad (77)$$

This way, the characteristic sensitivity value of each Extended CTC configuration will be calculated as the sum of the absolute maximum sensitivity values of all model parameters when an initial TCP pose deviation of $\delta \mathbf{x}_0 = 0.001$ m is considered.

Results are shown in Figs. 14 and 15. As it can be seen, the sensitivity of the Extended CTC approach varies depending on the sensor distribution, as seen previously. However, it can be seen that the dynamic performance and robustness of the use of redundant data is higher than the classical CTC approach (Config. 1).

The appropriate selection of the Extended CTC sensor configuration can be carried out applying different criteria. This way, if economic criteria are considered, a less number of sensors will reduce the cost of the robot. In these cases, Config. 3 and Config 4. seem to be the best alternatives considering the number of sensors vs performance criteria. However, if the robustness is to be maximized, the full sensorization of the robot (Config. 10) is the best alternative.

As it can be seen, it is shown that the characteristic sensitivity value calculated for the Extended CTC approach varies depending on the selected sensor configuration, illustrating the need of a preliminary analysis to select the most suitable sensor configuration for a given robot and task.

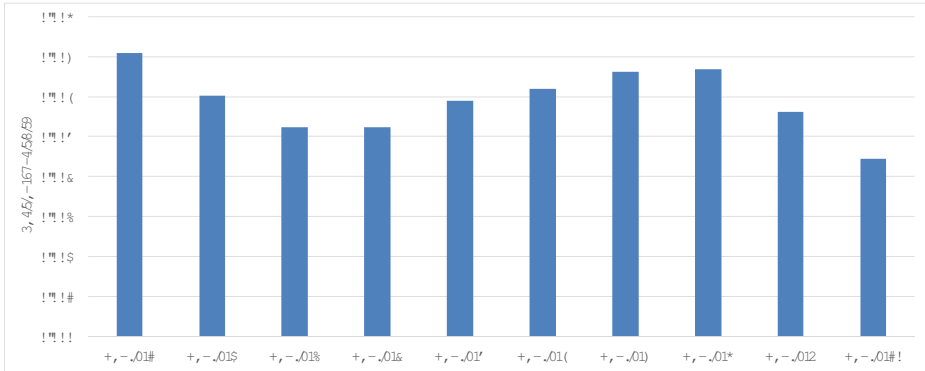


Figure 14. Normalized position Sensitivity $\sum \max \left| \xi_{\theta_k}^{POS} / \delta \mathbf{x}_0 \right|$.

Conclusion

Control is a central issue in parallel robos, as these mechanisms are usually used in high performance tasks where heavy load handling, precision and/or speed are required. However, being a recently rediscovered field, there are still many areas in parallel robotics that need further research.

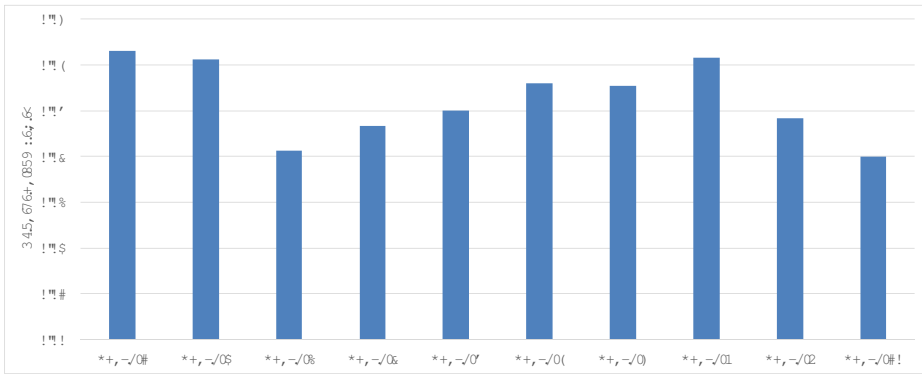


Figure 15. Normalized orientation Sensitivity $\sum \max \left| \xi_{\theta_k}^{ORI} / \delta \mathbf{x}_0 \right|$.

In this paper, the use of extra sensors in passive joints of parallel robots is proposed for position control purposes. The extra data provided by these sensors can reduce the computational cost of the kinematic and dynamic model, increase the robustness of the controller and provide better dynamic behaviour.

These advantages are demonstrated in the Extended CTC approach, a control scheme that combines the advantages of redundant sensors and advanced, model-based controllers. In this paper, the three steps for implementing this approach are analyzed briefly: the calculation of a generic, redundant, reconfigurable dynamic model, the definition of the control law, and its implementation. Moreover, a procedure to select the most suitable configurations is detailed based on a sensitivity approach. A study case based on the Gough-Stewart platform is proposed.

The procedures and control laws defined in this paper can be used in those applications where high performance criteria need to be met, such as medical applications or medium load pick-and-place tasks. Based on these ideas, new control approaches can be derived, modifying the formulation to consider sensors that measure other magnitudes such as acceleration or force. Further research is being carried on focusing in industrial applications such as test rigs for vehicle suspension based on parallel robots.

References

- [1] H. Abdellatif and B. Heimann. Computational efficient inverse dynamics of 6-dof fully parallel manipulators by using the lagrangian formalism. *Mechanism and Machine Theory*, 44:192–207, 2009.
- [2] J. Angeles. *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*. Springer, 2007.
- [3] L. Baron and J. Angeles. The kinematic decoupling of parallel manipulators using joint-sensor data. *IEEE Transactions on Robotics and Automation*, 16(6):644–651, 2000.
- [4] V. Bauma, M. Valášek, and Z. Sika. Increase of pkm positioning accuracy by redundant measurement. *Proceedings of the 5th Chemnitz Parallelkinematik Seminar*, pages 547–564, 2006.
- [5] S. Bhattacharya, H. Hatwal, and A. Ghosh. An on-line parameter estimation scheme for generalized stewart platform type parallel manipulators. *Mechanism and Machine Theory*, 32(1):79–89, 1997.
- [6] C. Brecher, T. Ostermann, and D.A. Friedrich. Control concept for pkm considering the mechanical coupling between actuator. *Proceedings of the 5th Chemnitz Parallel Kinematics Seminar*, pages 413–427, 2006.
- [7] P. Chiacchio, F. Pierrot, L. Sciavicco, and B. Siciliano. Robust design of independent joint controllers with experimentation on a high-speed parallel robot. *IEEE Transactions on Industrial Electronics*, 40(4):393–403, 1993.
- [8] A. Codourey. Dynamic modeling of parallel robots for computed-torque control implementation. *The International Journal of Robotics Research*, 17(12):1325–1336, 1998.
- [9] J.J. Craig. *Robotica*. Prentice Hall, 2006.
- [10] B. Dasgupta and T.S. Mruthyunjaya. A Newton-Euler formulation for the inverse dynamics of the Stewart Platform. *Mechanism and Machine Theory*, 33(8):1135–1152, 1998.

-
- [11] I. Davliakos and E. Papadopoulos. Model-based control of a 6-dof electrohydraulic Stewart Gough platform. *Mechanism and Machine Theory*, 43(11):1385–1400, 2008.
 - [12] F. Ghorbel, O. Chtelat, R. Gunawardana, and R. Longchamp. Modeling and set point control of closed-chain mechanisms: Theory and experiment. *IEEE Transactions on Control System Technology*, 8(5):801–815, 2000.
 - [13] H. Guo, Y. Liu, and H. Liu, G. ; Li. Cascade control of a hydraulically driven 6-dof parallel robot manipulator based on a sliding mode. *Control Engineering Practice*, 16(9):1055–1068, 2008.
 - [14] H.B. Guo and H.R. Li. Dynamic analysis and simulation of a six degree of freedom Stewart Platform manipulator. *Proceedings of IMechE Part C: J. Mechanical Engineering Science*, 220:61–72, 2006.
 - [15] M. Honegger, A. Codourey, and E. Burdet. Adaptive control of the Hexaglide, a 6 dof parallel manipulator. *IEEE International Conference in Robotics and Automation*, 1997.
 - [16] B. R. Hopkins and R. L. Williams II. Kinematics, design and control of the 6-PSU platform. *Industrial Robot: An international Journal*, 29(5):443–451, 2002.
 - [17] C-I. Huang, C-F. Chang, M-Y. Yu, and L-C. Fu. Sliding-mode tracking control of the stewart platform. *5th Asian Control Conference*, 1:562–569, 2004.
 - [18] W. Khalil and O. Ibrahim. General solution for the dynamic modeling of parallel robots. *Journal of Intelligent Robot Systems*, pages 19–37, 2007.
 - [19] D. H. Kim, Yang J-Y., and K-I. Lee. Robust tracking control design for a 6 dof parallel manipulator. *Journal of Robotic Systems*, 17(10):527–547, 2000.
 - [20] K.-M. Lee and D. K. Shah. Dynamic analysis of a three-degree-of-freedom in-parallel actuated manipulator. *IEEE Journal of Robotics and Automation*, 4(40):361–367, 1988.
 - [21] S-H. Lee, J-B. Song, W-C. Choi, and D. Hong. Position control of a Stewart Platform using inverse dynamics control with approximate dynamics. *Mechatronics*, 13:605–619, 2003.

- [22] D. Li and S.E. Salcudean. Modeling, simulation and control of a hydraulic stewart platform. *IEEE International Conference in Robotics and Automation*, 4:3360–3366, 1997.
- [23] E. Macho, O. Altuzarra, C. Pinto, and A. Hernández. Workspaces associated to assembly modes of the 5R planar parallel manipulator. *Robotica*, 36(3):395–403, 2008.
- [24] F. Marquet, O. Company, S. Krut, and F. Pierrot. Enhancing parallel robots accuracy with redundant sensors. *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, pages 4114–4119, 2002.
- [25] J-P. Merlet. Closed form resolution of the direct kinematics of parallel manipulators using extra sensors data. *Proceedings IEEE International Conference in Robotics and Automation*, pages 200–304, 1993.
- [26] J-P. Merlet. Still a long way to go on the road for parallel mechanisms. *Proceedings of the ASME 2002 DETC Conference*, 2002.
- [27] J. P. Merlet. *Parallel Robots (Second Edition)*. Kluwer, 2006.
- [28] R. Nakadate, H. Uda, H. Hirano, J. Solis, A. Takanishi, E. Minagawa, M. Sugawara, and K. Niki. Development of a robotic carotid blood measurement WTA-1RII: Mechanical improvement of gravity compensation mechanism and optimal link position of the parallel manipulator based on GA. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 717–722, 2009.
- [29] Y. Nakamura and M. Godoussi. Dynamics computation of closed-link robot mechanisms with nonredundant and redundant actuators. *IEEE Transactions on Robotics and Automation*, 5(3):294–302, 1989.
- [30] P.R Ouyang, W.J Zhang, and F.X Wu. Nonlinear PD control for trajectory tracking with consideration of the design for control methodology. *Proceedings of the 2002 IEEE International Conference in Robotics and Automation*, 4:4126–4131, 2002.
- [31] F. Paccot, N. Andreff, and P. Martinet. A review on the dynamic control of parallel kinematic machines: Theory and experiments. *The International Journal of Robotics Research*, 28(3):395, 2009.

-
- [32] V Parenti-Castelli and R Di Gregorio. A new algorithm based on two extra-sensors for real-time computation of the actual configuration of the generalized stewart-gough manipulator. *Journal of Mechanical Design*, 122(1):294–298, 2000.
- [33] I.T. Pietsch, M. Krefft, O.T. Becker, C. C. Bier, and J. Hesselbach. How to reach the dynamic limits of parallel robots? an autonomous control approach. *IEEE Transactions on Automation Science and Engineering*, 2:369–380, 1995.
- [34] J. Sabater, R. Saltarn, R. Aracil, E. Yime, J. Azorin, and M. Hernández. Teleoperated parallel climbing robots in nuclear installations. *Industrial Robot: An International Journal*, 33(5):381–386, 2006.
- [35] R Saltaren, R Aracil, P Cardenas, C Pena, E Yime, and C Alvarez. Underwater parallel robot for oceanic measuring and observations-remote: development and navigation control advances. *Proceedings of OCEANS 2009-EUROPE*, pages 1–9, 2009.
- [36] S. Stan, M. Manic, M. Maties, and R. Balan. Kinematics analysis, design, and control of an isoglide3 parallel robot (IG3PR). *Proceedings of the 34th Annual Conference of the IEEE Industrial Electronics Society*, page 12651275, 2008.
- [37] Y Su and B Duan. The application of the stewart platform in large spherical radio telescopes. *Journal of Robotic Systems*, 17(7):375–383, 2000.
- [38] Y. X. Su, B. Y. Duan, C. H. Zheng, Y. F. Zhang, G. D. Chen, and J. W. Mi. Disturbance-rejection high-precision motion control of a Stewart Platform. *IEEE Transactions on Control System Technology*, 12(3):364–374, 2004.
- [39] Y. X. Su, D. Sun, L. Ren, X. Wang, and J. K. Mills. Nonlinear pd synchronized control for parallel manipulators. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 1374–1379, 2005.
- [40] L-W. Tsai. *Robot Analysis*. Wiley-Interscience, 1999.
- [41] L-W. Tsai. Solving the inverse dynamics of a stewart-gough manipulator by the principle of virtual work. *ASME Journal of Mechanical Design*, 122:3–9, 2000.

- [42] M. Vukobratovic and M. Filipovic. Dynamic accuracy of robotic mechanisms. part 1: Parametric sensitivity analysis. *Mechanism and Machine Theory*, 35:221–237, 2000.
- [43] J. Wang and C. Gosselin. A new approach for the dynamic analysis of parallel manipulator. *Multibody System Dynamics*, 2:317–334, 1998.
- [44] C Yang, Q Huang, H Jiang, O Ogbobe Peter, and J Han. PD control with gravity compensation for hydraulic 6-dof parallel manipulator. *Mechanism and Machine Theory*, 45:666–677, 2010.
- [45] A. Zubizarreta, I. Cabanes, M. Marcos-Muoz, and Ch. Pinto. Control of parallel robots using passive sensor data. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008. IROS 2008*, pages 2398–2403, 2008.

Chapter 3

STRUCTURE SYNTHESIS OF FULLY-ISOTROPIC TWO-ROTATIONAL AND TWO-TRANSLATIONAL PARALLEL ROBOTIC MANIPULATORS

Yanbin Zhang*

School of Mechatronics Engineering,
Henan University of Science and Technology, Luoyang, China

Abstract

Strong kinematic coupling is one of inherent properties for general parallel robotic manipulators. Although it contributes high stiffness and large loading capability, it also leads to complicated kinematic solutions, difficult controlling design and path planning, which will take passive effects for the actual applications of the parallel manipulators. A new systematic method for structure synthesis of four-degree-of-freedom fully-isotropic parallel robotic manipulators is proposed in this paper. The moving platform of the mechanism consists of two-rotational and two-translational (2R2T) motion components with respect to the fixed base. Firstly, structure synthesis of uncoupled 2R2T parallel robotic manipulator is set up based on the concept of hybrid mechanism and the reciprocal screw theory for the first time. Then, type synthesis of fully-isotropic 2R2T parallel robotic manipulator is performed by changing one kinematical

* E-mail address: yanbin_zh@163.com; Address: School of Mechatronics Engineering, Henan University of Science and Technology, Luoyang, China.

chain's structure of the uncoupled parallel robotic manipulator obtained before. A lot of novel uncoupled and fully-isotropic 2R2T parallel robotic manipulators are synthesized. Kinematical Jacobian of a fully-isotropic 2R2T parallel robot is a 4×4 identity matrix, so there is a one-to-one linear mapping correspondence between the output velocity of the platform and the input velocity of the actuated joints. Therefore, one independent output of the moving platform can be controlled only by one actuator.

1. Introduction

A parallel robotic manipulator is a typically closed-loop mechanism which is composed of a moving platform connected to a fixed base by several serial open-single kinematical chains (or called limbs) in parallel [1-3]. In order to achieve certain special properties, the serial open-single kinematic chain was replaced by a hybrid chain [4, 5], in which there is at least one leg composed of a closed loop structure. Compared with traditional serial mechanisms, parallel manipulators have many merits, such as high stiffness, large load carrying capacity, high accuracy and high velocity. Therefore, parallel robotic manipulators have potential applications in the fields of industrial robots, flight simulators, parallel machine tools, medical devices, radar antennas and telescopes. Delta Robot [6] is one of the most famous parallel manipulators and applied successfully in the industrial fields. However, parallel manipulators also have some drawbacks, such as, small workspace, high coupling of kinematics and dynamics, difficult control design and path planning.

Four-degrees of freedom parallel robotic manipulators can be divided into three types based on the different motion output characteristics of the moving platform, i.e., one-rotational and three-translational type (1R3T), three-rotational and one-translational type (3R1T), two-rotational and two-translational type (2R2T). The former two types have been well studied and many novel manipulators were designed [7-9]. As for 2R2T parallel manipulators, only a few of them were reported in literatures. Lu [10] presented a 2R2T parallel manipulator with three legs, in which one of the legs provides two actuators to drive the platform. Abbasnejad [11] designed a 2R2T parallel manipulator with five limbs, in which one limb only restricts the output motion properties of the platform, but it does not control them.

Strong kinematic coupling is one of inherent properties for general parallel robotic manipulators. Although it contributes high stiffness and large loading capability, it also leads to complicated kinematic solutions, difficult controlling design and path planning, which will take passive effects for the actual

applications of the parallel manipulators. Recently, many scholars have paid more and more attentions on decoupled parallel robotic manipulators [12-17], which have simple linear kinematic solutions. The most remarkable characteristic for the decoupled parallel robotic manipulators is that their Jacobian is a triangular matrix. The mechanism is called uncoupled parallel robotic manipulator when its Jacobian is a diagonal matrix [18]. In this case, there exists a one-to-one relationship between the outputs of the moving platform and the inputs of actuators. If the Jacobian of a parallel manipulator is an identical matrix, it is defined as a fully-isotropic parallel robotic manipulator, because the condition number is constantly equal to 1 throughout the entire workspace. So this kind of manipulator has well performance in kinematic and dynamic transmission capabilities. Kong and Gosselin [19] addressed type synthesis of linear translational parallel manipulators based on the geometrical approach, among which some are uncoupled parallel manipulators and others are fully-isotropic parallel manipulators. Kuo and Dai [20] designed a fully-isotropic parallel orientation manipulator composed of three active legs and one passive spherical joint. Carricato [21] put forward type synthesis of fully-isotropic 3T1R parallel manipulators in terms of the screw theory. Planar 2T1R fully-isotropic parallel manipulators were investigated based on the screw theory and many novel manipulators were obtained in [22]. Gogu [23, 24] has completed type synthesis of fully-isotropic parallel manipulators based on linear transformation theory. Notably, structure synthesis of fully-isotropic 2R2T parallel manipulators has been performed as well [25]. However, each manipulator consists of one complex leg. This complicated structure probably leads to some problems in order to obtain the expected motion property and precision.

A new systematic method for type synthesis of the fully-isotropic 2R2T parallel manipulators is presented based on the concept of hybrid manipulator and the reciprocal screw theory in this paper. Other contents are arranged as following: structure design of a fully-isotropic 2R2T hybrid manipulator is discussed in section 2, there exists a one-to-one corresponding relationship between the output velocity of the moving platform and the input velocity of the actuated joints for these mechanisms. So each output motion of the platform is only controlled by one actuator.

2. Structure Design and Kinematic Analysis of a Fully-Isotropic 2R2T Hybrid Mechanism

A hybrid mechanism is defined as a mechanism which is composed of a parallel manipulator connected with a single open kinematic chain or two parallel manipulators assembled serially. In this paper, suppose each fully-isotropic 2R2T hybrid mechanism includes one parallel component and one serial component. The parallel part is a fully-isotropic 2T1R planar parallel manipulator selected from [22] and the serial part is a single open kinematic chain which only consists of one revolute joint. A new fully-isotropic 2R2T hybrid mechanism is shown in Figure 1. The parallel component of the hybrid mechanism is made of three legs. The first leg is composed of two prismatic pairs and one revolute joint. Their axes are perpendicular to each other. Thus, the structure of the first leg can be defined as $P_1 \perp P_2 \perp R_3$ from the base to the platform in sequence. The second consists of two prismatic pairs and three revolute joints. The axes of the first prismatic pair and the first two revolute joints are parallel to each other and perpendicular to the second prismatic pair, and the axis of the last revolute joint is perpendicular to the first prismatic pair, but neither parallel to nor perpendicular to the second pair, i.e., $P_4 // R_5 (\perp P_6 \perp) // R_7 \perp R_8$. The third consists of two revolute and two universal joints. The axes of two revolute joints are orthogonal. Adjacent rotary axes of two universal joints are parallel and the other axes of them are parallel to the axis of second revolute joint. The sequence of these joints from the base to the output link OL is $R_9 \perp R_{10} - U_{11} - U_{12}$. Here, P, R, C and U denote the prismatic, revolute, cylindrical and universal (or Hooke) joints, respectively. The assembly conditions of three legs are as follows: the axes of joints P_1 , P_4 and R_9 mounted on the base are normal to each other and these joints are selected as the actuated joints. Meanwhile, their axes are parallel to the x , y and z axes of the fixed frame $O-xyz$, respectively. Axes of joints R_3 and R_8 are collinear and perpendicular to the adjacent axis within U_{12} joint. Joints R_3 , R_8 and U_{12} are all connected with the output link OL . When three actuated joints are driven, the output link will translate along x -, y -axes and rotate around z -axis, i.e., the parallel part is a 3-DOF planar manipulator. Jacobian of this planar parallel manipulator is a 3×3 identity matrix [20]. In other words, this mechanism shows fully-isotropic throughout the entire workspace because the condition number of its Jacobian is identically equal to 1.

$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \\ \omega_v \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} \quad (1)$$

Where v_x and v_y are the linear velocities along x - and y -axes, respectively. ω_v and ω_z are the angular velocities around v - and z -axes, respectively. $\dot{q}_1$, $\dot{q}_2$, $\dot{q}_3$ and $\dot{q}_4$ are the generalized input velocities of four actuated joints. Obviously, the Jacobian is a 4×4 identity matrix, so this hybrid mechanism shows fully-isotropic too. There exists a linear mapping relationship between the input velocity space and the output velocity space.

3. Position Analysis

Referring to Figure 1, let point M is arbitrary on the moving platform and $\mathbf{M}_p = (r, 0, 0)^T$ is its coordinate with respect to the frame $P-uvw$. The coordinate of the origin point P within the fixed frame is $\mathbf{P}_o = (x, y, z)^T = (q_1, q_2, a)^T$, where q_1 and q_2 are the input displacements of the actuated joints P_1 and P_4 , respectively, a is the structure size denoting the distance from origin P to the fixed plane determined by x - and y -axes. The coordinate of point M with respect to the fixed frame can be calculated as follows

$$\mathbf{M}_O = \mathbf{R}\mathbf{M}_p + \mathbf{P}_O \quad (2)$$

where $\mathbf{M}_O = (x_M, y_M, z_M)^T$, $\mathbf{R}$ is the rotational matrix from the moving frame to the fixed one.

Assume θ and β are the posture angles of the moving platform. Since the hybrid mechanism is uncoupled, both posture angle θ and β are only relate to the actuated inputs q_4 of joint R_{13} and q_3 of joint R_9 , respectively. Thus, the rotational matrix $\mathbf{R}$ can be derived

$$\mathbf{R} = \begin{bmatrix} \cos \beta & -\sin \beta & 0 \\ \sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} = \begin{bmatrix} \cos \theta \cos \beta & -\sin \beta & \sin \theta \cos \beta \\ \cos \theta \sin \beta & \cos \beta & \sin \theta \sin \beta \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad (3)$$

Substituting Equation (3) into (2), yields

$$\begin{bmatrix} x_M \\ y_M \\ z_M \end{bmatrix} = \begin{bmatrix} r \cos \theta \cos \beta + q_{11} \\ r \cos \theta \sin \beta + q_{12} \\ r \sin \theta + a \end{bmatrix} \quad (4)$$

Based on Equation (4), one can see that x_M relates to the actuated input q_1 of the first leg and two pose angles θ and β of the platform. Similarly, y_M is the function of the variables of q_2 , θ and β . However, z_M only depends on the variable of θ , which means that if the position coordinate z_M is given, pose angle θ will be measured inversely. In other words, if there is another leg which is used to control coordinate z_M of point M independently, one uncoupled parallel manipulator will be obtained.

4. Structure Synthesis of the Fourth Leg

In order to design the uncoupled 2R2T parallel robotic manipulators, the fourth leg should be designed to provide single driving force for the platform along z -axis. Without loss of generality, suppose there are only two kinds of single-DOF joint in the fourth leg, i.e., prismatic and revolute joints. Structure synthesis of fully-isotropic translational parallel manipulators was addressed by use of the reciprocal screw theory in [26]. Research results show that the actuation screw, which provides the direct actuated force along one moving direction for the platform, of each leg for the fully-isotropic translational parallel manipulator, must be a pure force screw. At the same time, the actuated screw of the same leg should be an infinite-pitch screw and its axis is parallel to the moving direction.

Since the fourth leg is used to control the moving displacement of the platform along z -axis, the actuation screw $\$_{a4}$ should be a zero-pitch screw parallel to z -axis and the actuated screw $\$_4$ is an infinite-pitch screw parallel to z -axis [24], respectively. Therefore,

$$\$_{a4} = [0, 0, 1; P_{a4}, Q_{a4}, 0]^T \quad (5)$$

$$\$_4 = [0, 0, 0; 0, 0, 1]^T \quad (6)$$

where P_{a4} and Q_{a4} are the position parameters of $\$_{a4}$.

Equation (6) suggests that the actuated joint of the fourth leg is a prismatic pair and its moving direction is along z-axis.

In terms of the reciprocal law, the actuation screw is reciprocal to all kinematic screws of other joints with the exception of the actuated screw within the identical leg. Therefore, all the possible active non-actuated screws of the fourth leg can be listed below:

1. Zero-pitch screws parallel to z-axis. They cannot be directly connected to the moving platform or through the infinite-pitch screw. There must be at least one zero-pitch screw and at most three in the fourth leg.
2. Infinite-pitch screws perpendicular to z-axis. They can be placed at any position in the leg. There at most two infinite-pitch screws and their axes should not be parallel to each other.
3. Zero-pitch screw. It intersects the axis of the actuation screw $\$_{a4}$ and is parallel to the axis of joint R_{13} . There exists one and only one zero-pitch screw in the fourth leg.

In addition, the zero-pitch screw, which intersects the axis of the actuation screw $\$_{a4}$ and is perpendicular to the axes of the screws defined in steps (1) and (3), is called idle screw. This kind of screw can be inserted at any place and its number does not exceed one.

For the uncoupled 2R2T parallel manipulator, it is obvious that the platform has two rotational degrees of freedom. At the same time, three coordinates of point M are variable during the moving process, as shown in Equation (4). Therefore, the terminal link of the fourth leg should have at least three-translational and two-rotational degrees of freedom with respect to the initial link (or fixed base). If the connectivity of the fourth leg is equal to six, there must be one idle revolute joint. Otherwise, a redundant DOF exists.

According to the types and assembly conditions of the actuated joint and non-actuated screws, all potential kinematic chains of the fourth leg can be enumerated based on different connectivity (see Table 1). The first joint in each leg is selected as the actuator. As far as C joint connected to the base is concerned, its linear DOF will be chosen as the actuated input. Let subscripts, i , j and k , denote the translational or rotational directions of the corresponding joints and the unit vector space formed by three directions is linear independent. As for the P-pair following the subscript n , their moving directions must be perpendicular to u -axis. Let $\dot{X}$ represent a joint with an idle rotational DOF in the leg, where its connectivity is

equal to 6. In order to simplify the structure of the kinematic chains, the axes of the adjacent joints are regarded as parallel or perpendicular to each other. Such as the kinematic chain $C_i R_i \dot{U}_{ik} R_j$, the first joint is a cylindrical pair parallel to i-axis and its linear DOF is chosen as the actuated input, the second is a revolute joint parallel to i-axis too, the third is a Hooke joint with one axis parallel to the i-axis and another axis parallel to the k-axis, the rotational DOF about the k-axis is idle, and the last is a revolute joint parallel to the j-axis, which implies the last joint axis is normal to all others axes in the chain.

Table 1. Kinematic chains of the fourth leg

F_c	Type	Order	Basic chains	Chains with complex joints
5	1P4R	1-4	$P_i R_i R_i R_j$	$P_i R_i R_i U_{ij}; C_i R_i R_i R_j$ $C_i R_i U_{ij}$
	2P3R	5-16	$P_i P_n R_i R_i R_j; P_i R_i P_n R_i R_j$ $P_i R_i R_i P_n R_j; P_i R_i R_i R_j P_j$	$P_i P_n R_i U_{ij}; P_i R_i P_n U_{ij}$ $P_i R_i R_i C_j; C_i P_n R_i R_j$ $C_i R_i P_n R_j; C_i R_i R_j P_j$ $C_i P_n U_{ij}; C_i R_i C_j$
	3P2R	17-22	$P_i P_j P_k R_i R_j; P_i R_i P_k P_j R_j$	$P_i P_j P_k U_{ij}; C_i P_k P_j R_j$ $P_i R_i P_k C_j; C_i P_k C_j$
6	1P5R	23-47	$P_i R_k R_i R_i R_i R_j; P_i R_i R_k R_i R_i R_j$ $P_i R_i R_i R_k R_i R_j; P_i R_i R_i R_i R_k R_j P_i R_i R_i R_i R_j R_k$	$P_i \dot{U}_{ki} R_i R_i R_j; P_i R_i \dot{U}_{ki} R_i R_j$ $P_i R_i R_k R_i U_{ij}; P_i R_k R_i R_i U_{ij}$ $P_i R_i R_i \dot{U}_{ki} R_j; P_i R_i R_i R_k U_{ij}$ $P_i R_i R_i R_i \dot{U}_{kj}; P_i R_i \dot{U}_{ki} U_{ij}$ $P_i \dot{U}_{ki} R_i U_{ij}; C_i R_k R_i R_i R_j$ $C_i R_i R_k R_i R_j; C_i R_k R_i U_{ij}$ $C_i R_i R_i R_k R_j; C_i \dot{U}_{ki} R_i R_j$ $C_i R_i R_k U_{ij}; C_i R_i \dot{U}_{ki} R_j$ $C_i R_i R_i \dot{U}_{kj}; C_i \dot{U}_{ki} U_{ij}$ $P_i R_i R_i S; C_i R_i S$
	2P4R	48-77	$P_i P_n R_k R_i R_i R_j; P_i P_n R_i R_k R_i R_j$ $P_i P_n R_i R_i R_k R_j; P_i P_n R_i R_i R_j R_k$ $P_i R_k R_i P_n R_i R_j; P_i R_i R_k P_n R_i R_j$ $P_i R_i P_n R_k R_i R_j; P_i R_i P_n R_i R_k R_j$ $P_i R_i P_n R_i R_j R_k; P_i R_k R_i R_i P_n R_j$ $P_i R_i R_k R_i P_n R_j; P_i R_i R_i R_k P_n R_j$ $P_i R_i R_i P_n R_k R_j; P_i R_i R_i P_n R_j R_k$	$P_i P_n R_k R_i U_{ij}; P_i C_k R_i R_i R_j$ $P_i P_n R_i R_k U_{ij}; P_i P_n R_i \dot{U}_{ki} R_j$ $P_i \dot{U}_{ki} P_n R_i R_j; P_i R_k R_i P_n U_{ij}$ $P_i R_i C_k R_i R_j; P_i R_i R_w P_n U_{ij}$ $P_i R_w R_i R_i C_j; P_i \dot{U}_{ki} R_i P_n R_j$ $P_i R_i \dot{U}_{ki} P_n R_j; P_i R_i R_i R_i C_j$ $P_i R_i R_i P_n \dot{U}_{jk}; P_i R_i R_i C_k R_j$ $C_i P_n R_k R_i R_j; C_i R_k R_i P_n R_j$

If i-axis is parallel to z-axis of the fixed frame and j-axis is parallel to $O-xy$ plane, each kinematic chain in the Table 1 can be regarded as the fourth leg of a 2R2T parallel manipulator. It only provides the direct driving force for the platform along z-axis.

5. Structure Synthesis of Uncoupled 2R2T Parallel Robotic Manipulators

Uncoupled parallel robotic manipulator is a mechanism, in which a one-to-one corresponding control relationship exists between the output motion components of the platform and the inputs of the actuated joints. Each actuator of the mechanism controls one independent output motion component of the platform. An uncoupled parallel manipulator has the distinct characteristics that its velocity Jacobian is a diagonal matrix. Structure synthesis of uncoupled 2R2T parallel manipulators can be realized by use of one uncoupled 2R2T hybrid mechanism designed in Section 2 and one kinematic chain listed in Table 1. For example, one may choose the mechanism shown in Figure 1 and a $P_iR_iR_iR_iR_j$ chain in Table 1 to generate a new 2R2T parallel robotic manipulator named UPM2R2T-I (see Figure 2).

Some assembly conditions should be satisfied in order to achieve the uncoupled parallel robotic manipulator. Firstly, the axis of the actuated joint P_{14} located at the base is parallel to z-axis. Secondly, the axes of joints R_{13} and R_{18} attached on the platform are parallel to each other. In addition, the actuator used to drive the joint R_{13} in Figure 1 is removed, which means that the revolute R_{13} becomes a passive joint. The fourth leg controls the rotational DOF about the v-axis of the platform in Figure 2.

Referring to Figure 2, q_4 is the linear input displacement of the actuated joint P_{14} , r is the length of the platform's structural size. The velocity equation of point P on the platform within the fixed frame is

$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \\ \omega_v \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & b \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} \quad b = \frac{1}{r \cos \theta} \quad (7)$$

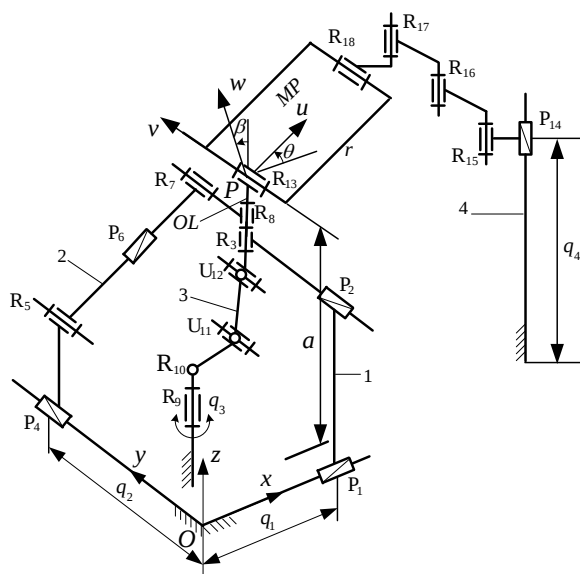


Figure 2. Uncoupled 2R2T parallel robotic manipulator UPM2R2T-I.

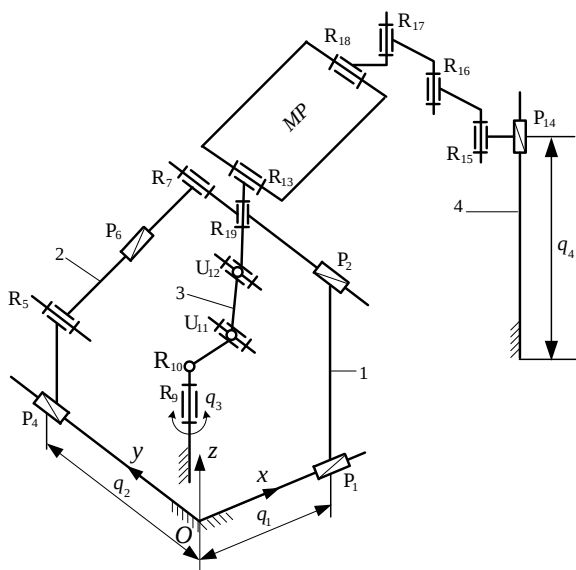


Figure 3. Uncoupled 2R2T parallel robotic manipulator UPM2R2T-II.

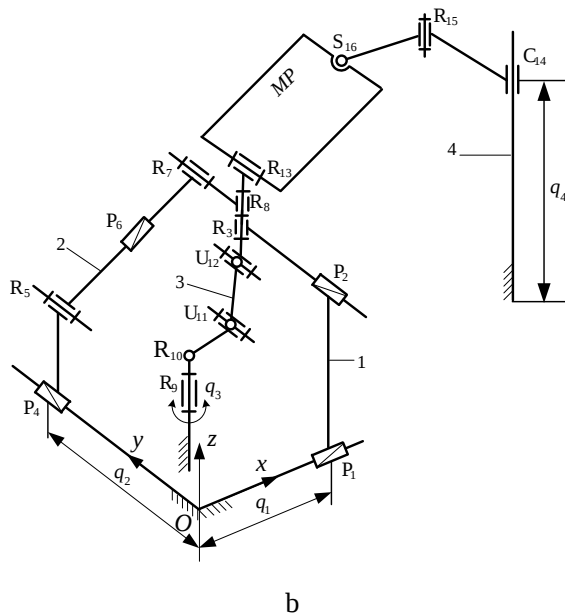


Figure 4. Other two novel 2R2T uncoupled parallel robotic manipulators.

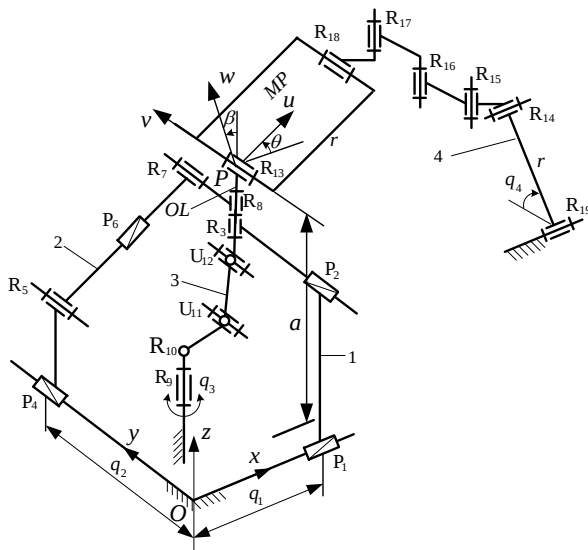


Figure 5. Fully-isotropic 2R2T parallel robotic manipulator.

6. Structure Synthesis of Fully-Isotropic T2R2 Parallel Robotic Manipulators

Replacing the prismatic pair of the fourth leg in UPM2R2T-I with a simple kinematic chain containing two revolute joints (R_{14} and R_{19}) with parallel axes will yield a new 2R2T parallel robotic manipulator, in which the axis of joint R_{14} (or R_{19}) is perpendicular to the axis of joint R_{15} and not parallel to joint R_{18} . In order to realize the expected target, there are some structural conditions to be satisfied. On the one hand, the plane defined by the axes of R_{13} and R_{19} is parallel to the $O-xy$ plane. On the other hand, link size between joints R_{19} and R_{14} is equal to the length of platform (see Figure 5). If joint R_{19} is selected as the actuated joint of the fourth leg and q_4 is the actuated input angle, then velocity equation of the verified manipulator in Figure 5 is as follows:

$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \\ \omega_v \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} \quad (8)$$

It is very interesting that the Jacobian in Equation (8) is a 4×4 identical matrix and its condition number is constantly equal to 1 throughout the whole workspace of the manipulator. This property reveals that the manipulator shows fully-isotropic in kinematics. So the mechanism is a fully-isotropic parallel robotic manipulator. Meanwhile, there is no singular configuration for the manipulator in the workspace since the special Jacobian.

According to above method, if the prismatic pair of the fourth leg in UPM2R2T-I is replaced by a parallelogram with four revolute joints and the side link length is equal to the platform size r , a new fully-isotropic 2R2T parallel manipulator is obtained (see Figure 6).

Similarly, for all uncoupled 2R2T parallel manipulators synthesized in Section 5, if the linear actuated joint in the fourth leg is replaced by a kinematic chain with two revolute joints or a parallelogram structure with four revolute joints, they can be evolved into fully-isotropic 2R2T parallel manipulators as well. It is true that the special structure size is necessary in order to obtain the expected

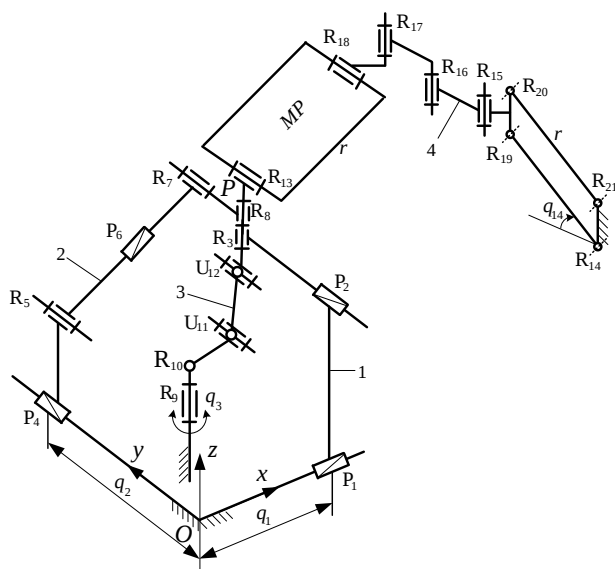


Figure 6. Another fully-isotropic 2R2T parallel robotic manipulator.

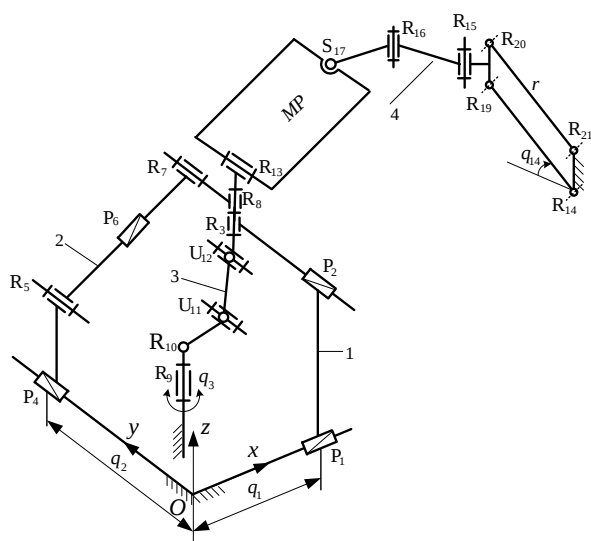


Figure 7. Novel fully-isotropic 2R2T parallel robotic manipulator with a parallelogram structure.

mechanisms. Furthermore, if the connectivity of the fourth leg is 6, its linear input joint can only be replaced by a parallelogram structure. Otherwise, the leg will include one redundant joint. For example, as far as the mechanism in Figure 5(b) is concerned, if the linear input DOF of the fourth leg is replaced by a parallelogram, a new fully-isotropic 2R2T parallel robotic manipulator can be developed (shown in Figure 7).

Conclusion

This paper presents a novel and systematic method for type synthesis of fully-isotropic 2R2T parallel robotic manipulators. The uncoupled 2R2T parallel robotic manipulators are synthesized based on the concept of hybrid mechanisms and the reciprocal screw theory. The fully-isotropic 2R2T parallel robotic manipulators are developed by using a kinematic chain with two revolute joints or a parallelogram structure to replace the linear input actuated joint in the fourth leg of each uncoupled 2R2T parallel mechanisms. For all of the proposed parallel mechanisms, their Jacobian that maps the input velocity vector space to the output velocity vector space of the platform is a 4×4 identity matrix. This leads to a one-to-one correspondence between the output motion components of the platform and the inputs of the actuators. Both the condition number and the determinant of the Jacobian matrix are equal to one. Therefore, there is no any singular configuration throughout the entire workspace and these mechanisms possess good performance characteristics in force and motion transmission. Control design and path planning of these mechanisms become very simple. All the uncoupled and the fully-isotropic 2R2T parallel manipulators proposed in this paper are original. Moreover, each parallel manipulator only consists of the prismatic, revolute, cylindrical, universal, or spherical joints, without the homokinetic joints or redundancy actuation. All these features will reduce the cost of manufacture and assembly.

Acknowledgments

The first author wishes to acknowledge the supports by the National Natural Science Foundation of China under grant 50905055, Program for Innovative Research Team (in Science and Technology) in University of Henan Province under grant15IRTSTHN008, and Program for Innovative Research Team of Henan University of Science and Technology under grant 2015XTD012.

References

- [1] Tahmasebi F. (2007). Kinematics of a new high-precision three-degree-of-freedom parallel manipulator. *ASME Journal of Mechanical Design*, vol. 129, pp. 320-325.
- [2] Zhu S. J., Huang Z. and Zhao M. Y., (2009). Singularity analysis for six practicable 5-dof fully-symmetrical parallel manipulators. *Mechanism and Machine Theory*, vol. 44, pp. 710-725.
- [3] Gallardo J., Orozco H. and Rico J. M., (2008). Kinematics of 3-RPS parallel manipulators by means of screw theory. *International Journal of Advanced Manufacturing Technology*, vol. 36, pp. 598-605.
- [4] Pierrot F., Marquet F., Company O. and Gil T., (2001). H4 parallel robot: modeling, design and preliminary experiments. Proceedings, IEEE International Conference on Robotics and Automation, Seoul, Korea, pp. 3256-3261.
- [5] Jin Q. and Yang T., (2004). Synthesis and analysis of a group of 3-degree-of-freedom partially decoupled parallel manipulators. *ASME Journal of Mechanical Design*, vol. 126, pp. 301-306.
- [6] Calvel R., (1988). DELTA, a fast robot with parallel geometry. Proceedings of the International Symposium on Industrial Robotics, Switzerland, pp. 91-100.
- [7] Kong X. and Gosselin C. M., (2004). Type synthesis of 3T1R 4-dof parallel manipulators based on screw theory. *IEEE Transactions on Robotics and Automation*, vol. 20, no.2, pp. 181-191.
- [8] Li Q. and Huang Z., (2003). A family of symmetrical lower-mobility parallel mechanisms with spherical and parallel subchains. *Journal of Robotic Systems*, vol.20, no.6, pp. 297-305.
- [9] Gallardo-Alvarado J., Rico-Martinez J. M. and Alici G., (2006). Kinematics and singularity analyses of a 4-dof parallel manipulator using screw theory. *Mechanism and Machine Theory*, vol. 41, pp. 1048-1061.
- [10] Lu Y., Hu B. and Shi Y., (2007). Kinematics analysis and statics of a 2SPS+UPR parallel manipulator. *Multibody System Dynamics*, vol. 18, pp. 619-636.
- [11] Abbasnejad G., Daniali H. M. and Fathi A., (2011). Architecture optimization of 4PUS+1PS parallel manipulator. *Robotica*, vol. 29, no. 5, pp. 683-690.
- [12] Zhang Y., Liu H. and Wu X., (2009). Kinematics analysis of a novel parallel manipulator. *Mechanism and Machine Theory*, vol. 44, pp. 1648-1657.

-
- [13] Briot S. and Bonev I. A., (2010). Pantopteron: a new 3T1R decoupled parallel manipulator for pick-and-place applications. *Mechanism and Machine Theory*, vol. 45, pp. 707-721.
 - [14] Zhang D. and Zhang F., (2011). Design and analysis of a totally decoupled 3-dof spherical parallel manipulator. *Robotica*, vol. 29, no. 7, pp. 1093-1100.
 - [15] Zeng D. and Huang Z., (2011) Type synthesis of the rotational decoupled parallel mechanism based on screw theory. *Science China technological Sciences*, vol. 54, no. 4, pp. 998-1004.
 - [16] Zeng D.-X., Hou Y. and Lu W.-J., (2014). Type synthesis method for the translational decoupled parallel mechanism based on screw theory. *Journal of Harbin Institute of Technology*, vol. 21, no.1, pp. 84-91.
 - [17] Yue C., Su H.-J. and Kong X., (2013). Type synthesis of 3-dof translational compliant parallel mechanisms. *Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference*, August, 2013, Portland, Oregon, US, DETC2013-12718.
 - [18] Bouzgarrou B. C., Dauroux J. C. and Gogu G. and Heerah Y., (2004). Rigidity analysis of T3R1 parallel robot with uncoupled kinematics. *Proceedings of the 35th international Symposium on Robotics*, Paris, France, pp. 1-6.
 - [19] Kong X. and Gosselin C. M., (2002). Type synthesis of linear translational parallel manipulators. *Advances in Robot Kinematics*, In: J. Lenarčič, P. Thomas (Eds.), *Advances in Robot Kinematics: Theory and Application*, Kluwer Academic Publishers, pp. 453-462.
 - [20] Kuo C. H. and Dai J. S., (2011). A fully-isotropic parallel orientation mechanism. *Proceedings 13th World Congress in Mechanism and Machine Science*, Guanajuato, Mexico, pp. 1-7.
 - [21] Carricato M., (2005). Fully isotropic four-degrees-of-freedom parallel mechanisms for Schoenflies motion. *International Journal of Robotics Research*, vol. 24, pp. 397-414.
 - [22] Zhang Y., Wang H. and Wu X., (2012). Structure synthesis of fully-isotropic 3-dof planar parallel manipulators. *Optics and Precision Engineering*, vol. 20, no.3, pp. 579-586.
 - [23] Gogu G., (2004). Structural synthesis of fully-isotropic translational parallel robots via theory of linear transformations. *European Journal of Mechanics A/Solids*, vol. 23, pp. 1021-1039.
 - [24] Gogu G., (2007). Structural synthesis of fully-isotropic parallel robots with Schonflies motions via theory of linear transformations and evolutionary

morphology. *European Journal of Mechanics, A/Solids*, vol. 26, pp. 242-269.

- [25] Gogu G., (2005). Fully-isotropic parallel robots with four degrees of freedom T2R2-type. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Alberta, Canada, pp. 960-965.
- [26] Zhang Y., Liu H. and Wu X., (2008). Type synthesis of non-singular full-isotropic translational parallel mechanisms based on theory of reciprocal screw. *Chinese Journal of Mechanical Engineering*, vol. 44, no. 10, pp. 83-88.

Biographical Sketch

Yanbin Zhang is a Professor at the School of Mechatronics Engineering, Henan University of Science and Technology, China. He received his PhD in Mechanical Engineering from Xi'an University of Technology in 2008. His research interests include mechanisms, parallel robotic manipulator, mechanical design theory. He has published 30 academic papers in journals such as *Mechanism and Machine Theory*, *Int. J. Advanced Robotic Systems*, *Journal of Mechanical Engineering* (in Chinese), *Optics and Precision Engineering* (in Chinese), *China Mechanical Engineering* (in Chinese), *Transaction of the Chinese Society for Agricultural Machinery* (in Chinese), etc.

Chapter 4

THE PARALLEL TWO-LEGGED WALKING ROBOT CENTAUROB

Shucen Du, Josef Schlattmann, Stefan Schulz and Arthur Seibel*

Workgroup on System Technologies and Engineering
Design Methodology, Hamburg University of Technology,
Hamburg, Germany

Abstract

In this chapter, we review the new prototype of the two-legged, parallel kinematic walking robot CENTAUROB, developed at Hamburg University of Technology. The main features of this robot are its modular, symmetric structure, the construction of the legs as Stewart platforms (hexapods), and the special C-shaped feet that allow a statically stable movement at any time. The symmetric structure of the robot leads to a center of gravity exactly in the middle of the hip and prevents from a preferred moving direction or an asymmetric load of the legs. As a highly flexible walking device, the CENTAUROB is able to walk in unpaved environment, avoid and overcome obstacles, and even handle straight and circular stairs.

Keywords: parallel robots, service robots, biped walking, statically stable walking, sensor architecture, control concepts, walking scenarios

*E-mail address: st.schulz@tuhh.de (Corresponding author).

1. Introduction

Industrial robots are well established in production processes. Since recent years, there has also been a growing demand for service robots both for private use, for example, as a ride-on device for the mobility assurance of immobile people or as an interactive care giving robot in the household, and for use in industrial environments, for example, as a simple industrial carrying device or as an autonomous robot in rough or dangerous areas like mine fields or burning houses. Market studies show that the demand in robotic systems will rise steadily in the coming years. Due to higher performance-requirements with respect to dynamics, precision, velocity, weight, control intelligence, suitability for automation, price, etc., most of the conventional robot solutions have reached their limits.

Biped robots can be classified by their kinematic structure into serial and parallel ones. Serial kinematic robots are most often used due to the lower control effort. Well known examples for humanoid robots with a serial kinematic are Honda's humanoid robot Asimo [1], the humanoid robot LOLA [2], and the WABIAN series [3]. Parallel robots offer special advantages including higher stiffnesses, smaller overall machine sizes, lower inertial forces, faster processing velocities, and compensatory dimensional errors. The operating load occurring at the end effector is divided among several links which results in a high stiffness of the structure. The two-legged, parallel kinematic walking robot CENTAUROB, developed in 1997 [4], was by then the first biped robot using two Stewart platforms as leg structures. It was the first parallel walking robot that was able to walk statically stable even in unpaved environments, avoid and overcome obstacles, and even handle straight and circular stairs. A second realization of a biped robot with a parallel kinematic, which exists up to now, is the Waseda-Leg [3].

In this chapter, we review the new prototype of the CENTAUROB [5], developed at Hamburg University of Technology. Just like the first prototype [6], the redeveloped construction is based on the Stewart platform as well. Several modifications were made due to the moving technique standard and to solve disadvantages and problems [7].

The new prototype represents a universal moving device which convinces by its high stability and payload and offers plenty of application possibilities. In future, the robot will be able to operate autonomously and automatically recognize obstacles and overcome them. Possible fields of usage that come into mind are, for example, as a ride-on device for immobile people, as a service or

interactive care giving robot in the household, as a simple industrial carrying device in unpaved terrain, or as an autonomous robot in rough or dangerous areas like mine fields or burning houses.

The remainder of this chapter is organized as follows. In Section 2, we discuss parallel robots and their applicability as biped walking robots. In Sections 3 and 4, we review the first prototype, the redesign process, and the new prototype of the CENTAUROB. In Section 5, we briefly present the inverse kinematic model of the robot followed by the motion control strategy in Section 6. Section 7 presents several walking scenarios and other motions to show the agility of the robot. In Section 8, possible applications and fields of usage for the robot are discussed. Finally, in Section 9, this chapter is summarized and an outlook to future work is given.

2. Parallel Walking Robots

2.1. Parallel Robots

Currently, the field of industrial robots is dominated by articulated robots where the driven axes are arranged in series, so-called serial robots. A well known example is the SCARA robot, short for 'Selective Compliance Assembly Robot Arm'. Serial robots owe their success mainly to the high degree of flexibility, the high workspace-to-footprint ratio, as well as the low calculation and control effort due to the easily solvable direct kinematics. Although the serial structure leads to high flexibility, this is also a main disadvantage since positioning errors in the driven joints as well as the elasticity of each component of the structure accumulate and lead to lower positioning accuracies and vibration problems. In order to compensate these problems with higher driving forces and inertia torques, drives that are further away from the end-effectors are scaled larger.

Parallel robots can overcome these drawbacks. The name results from their parallel kinematic structure in which the drives operate side by side. The load is distributed over multiple paths, and the drives do not have to hold each other. This results in high rigidity, similar to a framework. The positioning accuracy is also very high because the positioning errors of the individual drives do not accumulate. The moving masses are smaller, so that the machine dynamics can be controlled by the use of lower power ratings. As a result of these advantages, parallel robots have become more relevant in fields where high process forces and special accuracies are required. The main applications for parallel robots

are widely ranged and spread from motion simulators and test benches over machine tools for handling and manufacturing to brain surgery. However, the mobility and flexibility of serial robots cannot be obtained.

There are many possible configurations of parallel robots which differ, in particular, by the type of drive (rotary or linear) and the degrees of freedom. One of the most universal parallel robot is the hexapod. The hexapod is also called Gough-Stewart platform according to the developers who have used these parallel kinematics for the first time [8], [9].

The function of a hexapod is to move the end-effector in all translational and rotational axes in space. Six parallel drives connect the stationary base platform with the mobile tool platform. On top of the tool platform, the end-effector is mounted for specific tasks. If, for any particular purpose, it is not necessary to reach the full dimension of the workspace, the number of drives can be reduced as long as additional joints or supports are introduced. For example, when using a rotary tool, such as a milling cutter, no rotatory degree of freedom is necessary. A well known example of a parallel manufacturing robot is the three degree of freedom delta robot [10], which uses only three drives and an additional passive guidance.

2.2. Service Robots

Generally, biped robots will take their place in the near future in the area of mobile service robots and will act as a substitute for humans or support them. The tasks will range from the operation in hazardous environments up to a general support in the household [11]. Especially in Japan, the realization of this vision is already well advanced. In the field of service robotics, a dichotomy between specialized and universal robots can be observed. Specialized service robots already have a high practical relevance and were tested for various distribution functions, for example, in hospitals [12], [13], [14]. Here, the drive kinematics has minor importance since navigation is only necessary in known barrier-free environments. In addition to these robots, for specific tasks, there is the aim to construct a universal robot. Generally, a human-like form is used as a model.

According to the definition of the International Federation of Robotics, a service robot is described as follows. 'A service robot is a robot which operates semi- or fully autonomously to perform services useful to the well-being of humans and equipment, excluding manufacturing operations.'

Currently, autonomy is the biggest problem. Further challenges are the

power supply and the autonomous navigation. By methods of image recognition and processing, robots are independently able to move. The energy consumption of the drives, however, still leads to short periods of use. Optimally, battery charging stations are installed that are automatically visited by robots with a low energy level. The wider the field of applications get, the larger and heavier become the batteries the robot has to carry. Especially for humanoid robots, this is a huge problem because they usually carry the batteries as a backpack. With increasing weight of the batteries, the center of mass is shifted upwards, resulting in a lower tilt stability.

2.3. Humanoid Robots

In order to construct a robot, the human locomotive system is often used as a model. These robots are called humanoid robots. Such a transfer of biologically successful principles into a technical application is called bionics.

The motion of a biped is one of the major challenges of research because it allows optimal locomotion in unstructured environments. Following the guiding principle of bionics according to the model of joint and muscle arrangements of humans, a robot with a comparable consolidation achieves an approximately equal mobility. An essential feature of the construction of the legs are the joints at the hips, the feet, and especially the knees. Compared to human joints, they are limited in the degrees of freedom, number, size, and resilience. As a result, only a simplified transfer to the robot is possible. However, it should be noted that bionics is only implicitly mentioned as a motivation for the design of a robot with knee joints. Equally important is certainly the vision of maintaining or increasing the workforce of a human by a robot, which is optimally solved when the robot is built like a human. Moreover, the acceptance of a humanoid robot would probably be higher since it would act more as a person and less as a machine.

The biggest problems of bipeds are balance control and coordination of motions, which have already been treated in many works. Even for well-developed models, fast walking is still challenging. This fails mostly due to the high computation burden. The best-known representatives of humanoid robots are ASIMO [1], the model of the company Honda, and LOLA [2], a very advanced model with automatic obstacle detection from the Technical University of Munich.

The overall conclusion is that many research groups and institutions are

concerned with the research and development of humanoid robots. Aside from a few models of large corporations, only a few bipeds have left the prototype stage yet. The reason is, probably, the relatively low interest of the industry since the way to a fully operational, robust, and secure biped is still far away. ASIMO, for example, which has already achieved a worldwide recognition, is more used to serve the image of the company than to execute household's duties. However, the theoretical fields of usage and applications for humanoid robots are very large, for example, in dangerous environments. Here, bipeds could be invaluable, so that their development is of great benefit.

2.4. Alternative Leg Structures

Usually, the joints of the robot are actuated directly. The stabilizing effect of muscles, which are connected to the bone, is not given. The problem of the lower stiffness of serially robots thus transfers to the robot's knee. This problem can probably be solved constructively or by reducing the walking speed. As a reasonable alternative, parallel kinematics can be used which offer high stiffness and stability. However, the flexibility and the large workspace of serial kinematics cannot be reached because the stability allows only small walking steps.

3. First Prototype of the CENTAUROB

The first prototype of the CENTAUROB, shown in Fig. 1, consists of two hexapods as leg structures. This feature is a fundamental difference to other humanoid robots. It establishes new ways of movement strategies and tasks. The aim is to use the robot in specific fields where the advantages of its parallel kinematic structure can be fully exploited. These advantages mainly include the high payload capacity, the high rigidity, as well as the possible large step size and step height. High steps and staircases can be climbed by alternately extending and retracting the telescopic legs. Moreover, passing low openings is also not a problem because both legs can be simultaneously retracted to reduce the body height.

The robot has two specially designed C-shaped feet to ensure a statically stable walking at any time. This means that the ground projection of the overall center of mass of the robot must always remain in the area spanned by the feet,

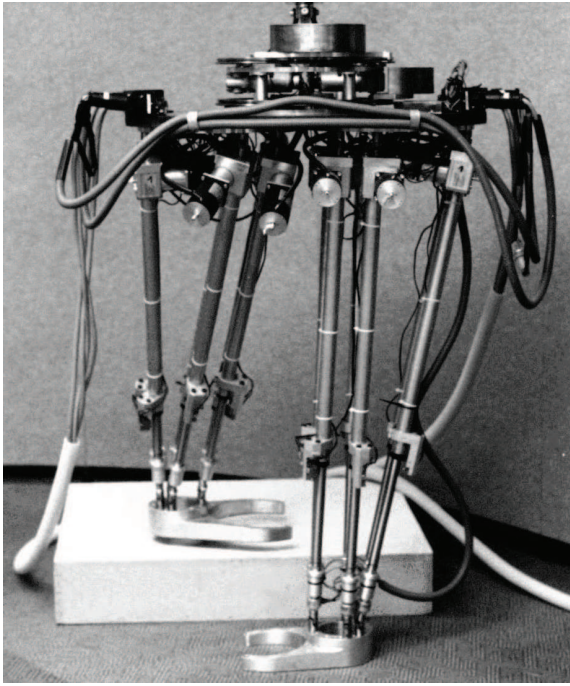


Figure 1. The world's first spiral stair climbing robot, own patent.

the so-called stability region. Figure 2 shows the robot's stability region with one foot and two feet on the ground.

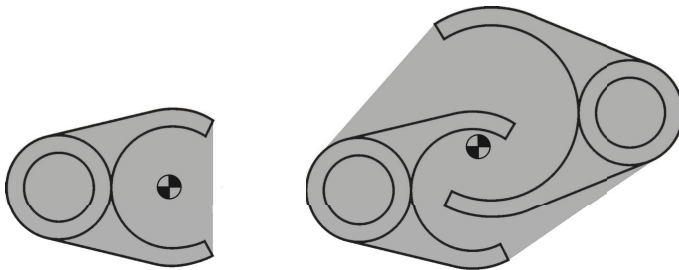


Figure 2. Stability region of the CENTAUROB with one foot (left) and two feet on the ground (right).

The joints of each leg are arranged in circles with different radii. A reduction

of the upper joints' radius would move the static and kinematic properties of the legs to an unfavorable area. The requirements for small space and high stability are divergent and cannot be optimally achieved simultaneously.

The robot is operated by a pure feed-forward or open loop control. Because for statically stable motion, dynamic influences such as accelerations are disregarded, the robot's movement can only be performed very slowly.

The CENTAUROB's advantages and disadvantages compared to other humanoid robots are summarized in Table 1.

Table 1. Advantages and disadvantages of the CENTAUROB compared to other robots

Advantages	Disadvantages
high payload to weight ratio	limited workspace due to leg collisions
high feet flexibility	operating close to singular configurations
ability to overcome complex obstacles	large leg cross sections
statically stable motion	high calculation effort in control
accurate and stable alignment of the hip while standing	friction afflicted operation of all actuators for the movement in only one degree of freedom
retracting and extending of legs	
leg structure modularization	
failure save system	

3.1. Structural Information

3.1.1. Workspace

Investigating the kinematics of a robot, the workspace is of great interest. Here, a distinction is made between three different workspaces:

- The *maximum workspace* indicates the points that can be achieved with

at least one orientation of the platform.

- The *total orientation workspace* is a subspace of the maximum workspace and indicates the points that can be achieved in any orientation of the platform.
- The *dexterous workspace* indicates the points that can be achieved at a given fixed orientation of the platform.

For hexapods used as machine tools, the calculation of the workspace plays an important role. This leads to great interest in capturing tool shifts or inclinations. The large orientation angles mark out the area of operation of a machine and prove its competitiveness. In case of the CENTAUROB, however, these extreme positions are never approached because the robot might lose its stable state.

For each leg of the CENTAUROB, the calculation of the workspace describes the positions and orientations the corresponding foot can reach in a single step. Investigating only straight planes, the calculation of the dexterous workspace is sufficient. Here, it can be assumed that the orientation of the foot platforms are always horizontal, respectively, parallel to the floor. Climbing stairs is also covered in this case. In addition, studies can be carried out with inclined feet. However, this scenario hardly shows significant changes in most of the parameters. It is related to the elongated shape of the hexapod, where the orientation of the foot platform has low impact on the length and inclination of the individual drives. The dexterous workspace of the CENTAUROB is illustrated in Fig. 3.

With the help of the calculated workspace, two parameters can be derived. These are the step height and the step size of the leg. Particularly, for determining the maximum surmountable obstacle height, the step height is important. This applies even more to the CENTAUROB as its telescopic leg kinematics leads to a special mobility. The step height is derived from the vertical workspace limits for $x = 0$ and $y = 0$. This corresponds with the maximum retracted and extended leg in a horizontal state. Is the vertical position known, the possible step size can be determined from the boundary of the workspace.

3.1.2. Singularities

The workspace specifies only the constructive constraints. Whether the drives can reach a specific point or not, is not described by the workspace. Singularities

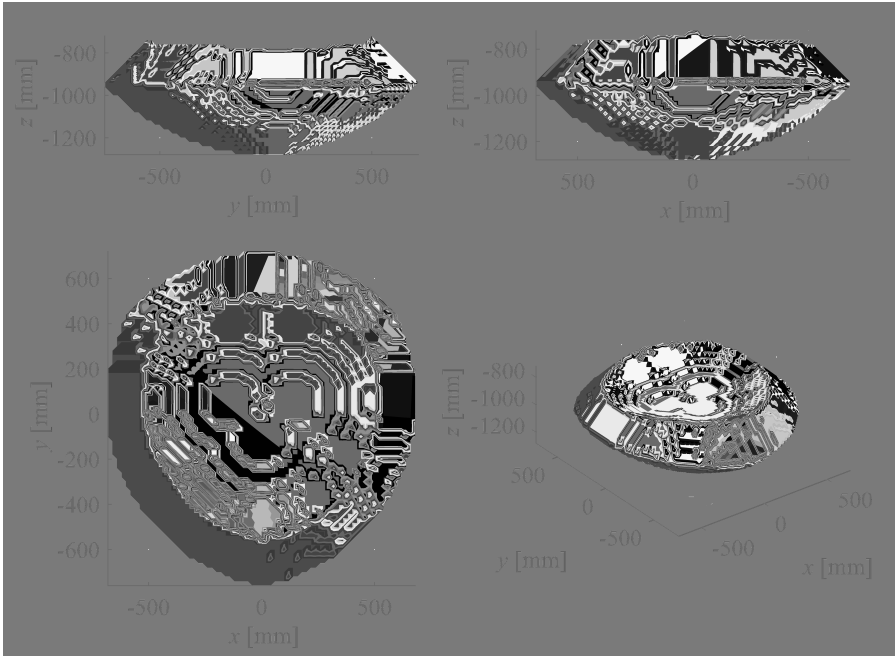


Figure 3. Dexterous workspace of one leg of the CENTAUBOB.

can occur at points inside the workspace and cause severe problems. Mathematically, a singularity is a point at which a model is not defined or actually does not work. In a singular configuration, the system's behavior is not predictable and the corresponding motions and forces become uncontrollable. Singularities and even configurations close to the singularities should be avoided due to the severe effect on the input-output transmission and possible failures or permanent damages.

The case of losing a degree of freedom occurs, for example, when the drives shall act in a perpendicular direction to the drives. In this direction, no forces can be applied, with the result that a virtual displacement is possible. This is a critical situation for a hexapod. In particular, the extended structure of the CENTAUBOB nearly leads to parallelly arranged actuators that can take forces only slightly in horizontal direction.

Degrees of freedom can be obtained when a movement of the structure is possible for fixed positions of the end-effector. This typically occurs in serial

robots when joint axes of drives coincide in the structure. Then, for example, the total rotation angle is known while the single rotation angles cannot be uniquely determined. The occurrence of singularities can be observed in the kinematic model when the JACOBI matrix loses rank.

3.1.3. Stability

Stability, especially tilt stability, has high priority for legged robots. It is an important part in mainly all phases of the robot's movement and can be directly influenced by the structural design, especially of the feet, and the walking pattern. Interferences, such as uncertainties in the environment or calculation errors, shall not unbalance the robot. Both are grouped under the term stability.

If a foot is placed on the ground, a contact situation occurs. In this contact situation, tangential and normal forces as well as moments act on the foot. They result from internal and external forces on the robot's body. For the foot, there exists a point in which the moments in the plane of contact accumulate to zero. This zero-moment point is consistent with the point where the resulting compressive force acts on the foot. The stability of the robot is ensured if the zero-moment point lies within the stability region, which is formed by the convex hull of all points of the contact surface.

For the CENTAUROB, a statically stable walking pattern is embedded. This means that dynamic influences are neglected and only slow, quasi-static movements are possible. Therewith, only the weight of the robot influences the robot's stability. Static stability is thus given if the ground projection of the robot's center of mass lies within the area formed by the feet in contact with the ground, see Fig. 2.

3.1.4. Agility

The locomotion of the CENTAUROB mainly depends on the relative movement between the hip and the foot platform. The transmission of movements from the drives to the platforms is given by the kinematics, see Section 5. Therewith, the maximum platform velocities can be determined from the maximum velocity of a single drive.

For serial robots, the maximum velocities can usually be determined directly from the performance of the drives, as these are each associated with one degree of freedom alone. The velocities of an industrial robot cannot be achieved with

the CENTAUROB. However, these velocities are not necessary, as long as a statically stable movement is required.

3.1.5. Stiffness

In real environmental situations, static and dynamic excitations can act on a robot. A static excitation can be, for example, a load that is carried by the robot. Impulses that occur when placing the foot on the ground lead to dynamic excitations. Further impulses can occur in case of interactions with objects. The stiffness of the leg structure determines how it reacts on these excitations. By using a hexapod, the robot's stiffness is inherently higher than the stiffness of humanoid robots with knees.

The stiffness is determined by a global stiffness matrix, which is derived from the resilience of the drives. Since the drives are connected to universal joints at their ends, no bending moments appear. The purely axial loads allow to model the drives as tie bars or struts. Here, the elasticities of further components are not considered explicitly. Instead, they are considered by a high safety factor. From the diagonal elements of the stiffness matrix, the rigidities can be determined in every direction of the world coordinate system, and the influence of off-diagonal elements is disregarded.

4. New Prototype of the CENTAUROB

4.1. Redesign Process

The first prototype of a parallel two-legged walking robot was a breakthrough in technology and base for further engineering. The vision of creating a mobile robot which can operate autonomously and automatically recognize obstacles as well as overcome them implies using an independent power supply, a stand-alone control system, and compliance to essential safety requirements in human surroundings. For this reason, the CENTAUROB underwent a complete redesign in which every mechanical component was thoroughly reviewed to reduce weight as well as increase dynamics and efficiency [7].

As a first step in the redesign process, an optimization of the Stewart platform architecture was suggested. The circular shape of the Stewart platforms attached to the hip was changed to an elliptical shape with a smaller diameter only in the lateral direction, see Fig. 4. Therewith, the lateral dimension of the

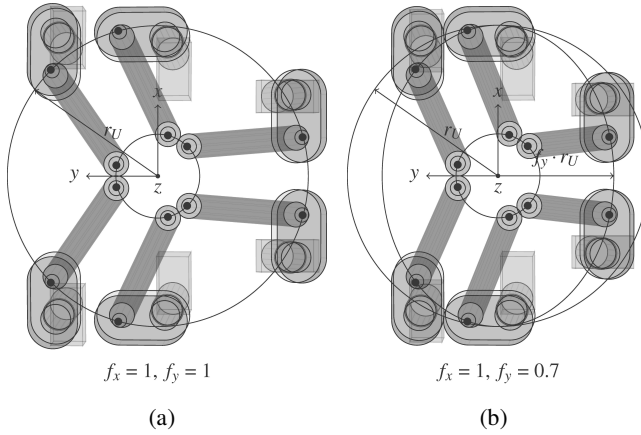


Figure 4. Redesign of the robot's hip: (a) original, circular hip, and (b) optimized, elliptical hip with reduced lateral dimension.

hip can be reduced to a more human-like one. Furthermore, the width of the hip determines the flexibility of the robot as it limits the opportunity to pass through narrow passages. By this means, the good performance in the dominant forward direction can be preserved in combination with a narrow hip size. Other changes in the mechanical components include a replacement of the formerly used stepper motors by high performance electronically commuted servomotors and the sliding screws by low-friction ball screws, which require brakes in case of a power or control failure since the new mechanism is not self-locking. Due to the low coefficient of performance, instead of worm gears, we now use belt drives and planetary gears between the spindles and motors. In order to compensate the passive rotation of the upper cylinder around the spindle axis [15], an upper joint with an extra degree of freedom of rotation is possible. A more sophisticated possibility is a compensation of the passive rotation by using quaternions as proposed in [16]. Therewith, universal joints can be used for the upper and lower joints as well. The new prototype of the CENTAUROB is illustrated in Fig. 5.

4.2. Structural Design of the New Prototype

The main modules of the CENTAUROB are the hip, the upper joints, the legs, the lower joints, and the feet. The main components of the robot are briefly in-

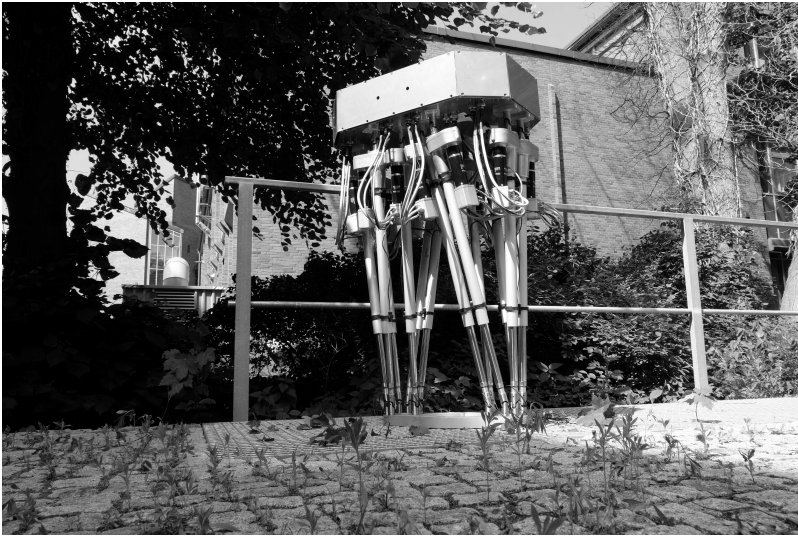


Figure 5. New prototype of the CENTAUROB.

roduced in the following. A more detailed description of the robot's mechanical structure is given in [5].

4.2.1. Hip

The hip, which is the connecting structure between the legs of the robot, contains the motor cards, the power supply, and the on-board control system. The size of the hip is 720 mm in the lateral direction and 500 mm in the longitudinal direction and is adapted to the positions of the upper joints. The upper joints of each hexapod are arranged side by side in an ellipse instead of a circle to provide a narrow hip size as described in Section 4.1.

4.2.2. Joints

As upper and lower joints, we use state-of-the-art universal joints made of stainless steel with two degrees of freedom and a pivot angle of 45° . In this case, the passive rotation that results from the asymmetric transformation of rotations in the universal joints, must be mathematically formulated and computationally compensated [15], [16].

4.2.3. Legs

The legs of the robot are composed of six linear units that are combined to a hexapod. Each linear unit itself consists of a ball screw linear actuator and a parallelly arranged motor unit. Both are connected by a cam belt.

The motor unit consists of an electrically commutated DC motor, a motor card, which is placed on the hip, a brake, and a planetary gear. The motor card allows a position, velocity, and current control. Furthermore, the motor contains Hall sensors and a three-channel encoder to detect the angular position of the motor shaft and therewith the length of the linear actuator. The brake is used to hold the position in case of an electricity failure, and the interposed planetary gear leads to a higher torque in the linear actuator.

The cam belt transmits the motor's torque one-to-one to the linear actuator. Then, the ball screw converts the torque into a linear movement. The linear actuator consists of an upper cylinder with an integrated spindle and a lower cylinder, which can move in linear direction. Strain gauge force sensors are used at the end of each linear unit for measuring the forces along the spindle.

4.2.4. Feet

The special C-shape of the feet leads to a convex standing surface containing the ground projection of the robot's center of mass, see Fig. 2. The hip of the robot can be moved as long as the ground projection of the center of mass remains in the standing surface. Due to the special shape of the feet, it is also possible to place the smaller foot inside the bigger foot. When the feet are placed inside each other, the height of the CENTAUROB from the ground to the bottom edge of the hip is 872 mm in the retracted and 1269 mm in the extended condition.

4.3. Sensor Architecture

The global sensor concept for the CENTAUROB contains the so-called pose detection, the force-measurement, and the detection of the ground projection of the robot's center of mass.

4.3.1. Pose Detection

Six minimal coordinates are necessary to determine the pose of a six degree of freedom parallel mechanism. In order to determine the pose and orientation of such a mechanism, generally, the lengths of the six linear actuators are

measured. This so-called direct kinematics problem is quite difficult to solve because there is usually not a unique solution. It is necessary to search iteratively for the platform's positions and orientations with the need of an initial pose estimate. The accuracy and convergence time depends on the choice of this initial pose estimate. Many studies have been made in this field. In order to find a unique solution for a given set of actuator lengths, it is possible to add further sensors on the passive joints of the hexapod [17], [18], [19], and others. Two possibilities of the placement of sensors have been investigated. One possibility is to add rotary sensors on existing passive joints, and the other possibility is to add passive links whose lengths are measured with linear sensors.

For an electric driven linear actuator, it is indeed possible to detect its length with a rotary encoder. Rotary encoders provide precise values of the angular position of the motor shaft, which, in turn, is related to the linear actuator's length. This length, on the other hand, needs to be linked with the encoder position by performing an initial length measurement. Due to a possible backlash in the gear or the spindle, the flexibility of the belt, or the so-called passive rotation in unguided ball screw linear actuators, the obtained data do not have to comply with the actual linear actuator's lengths necessarily [15], [16], [20]. An additional compensation or a numerical real-time estimator, as suggested in [20], needs to be implemented.

Although the lengths of the linear actuators are needed, for example, as target values for the motors, the position and orientation of the platforms are more valuable. One reason for this is, for example, the use of these values as a reference signal in control. Furthermore, using inverse kinematics (see Section 5), the linear actuator's lengths can be obtained from the platform's positions and orientations.

As mentioned before, for incremental length sensors, an initial length measurement or a so-called homing, an initialization run to predefined reference signals, is necessary. For absolute length sensors, the accuracy decreases with increasing measuring range. Furthermore, the installation range and the cost also increase.

For the two-legged walking robot CENTAUROB, several sensor concepts have been studied. Four sensor concepts using only the minimum number of sensors were proposed and presented in detail in [21]. These concepts still need further numerical methods to find the unique pose of the robot. They are described in the following.

Six angle sensors are mounted on the six linear actuators of the hexapod to detect the orientation of each linear actuator. Therewith, manufacturing costs can be reduced because the linear actuators can be constructed identically. A problem here is that even small angle changes can result in major changes in the linear actuators' length.

Six displacement sensors are mounted on the six linear actuators of the hexapod. These sensors detect length changes in the ball screws of the linear actuators. In this context, different types of displacement sensors can be implemented. For example, it is possible to implement displacement sensors where the measuring armature dips into a core, a magnet is moved along a track, or the length is detected with a rope. The advantage of using displacement sensors is the high and uniform accuracy of the measurement. Even for simple executions, these measuring units achieve an accuracy of up to 0.25 mm. This accuracy is approximately constant for the entire stroke of the linear actuator. Moreover, displacement sensors can directly detect length changes of the linear actuators. The main disadvantage of displacement sensors, however, is that most of them are quite large and must be mounted directly on the ball screw linear actuators.

Three angle sensors and three displacement sensors are mounted on three linear actuators to combine the advantages of both sensor concepts. Thus, the lengths of the linear actuators can be measured with displacement sensors and the orientation of the foot can be determined by angle sensors. Thereby, the space required for the sensors is minimized so that the workspace is only as limited as possible. In particular, due to the serial arrangement of the two hexapods of the CENTAUROB, the workspace of the legs is already limited. In order to achieve an optimal sensor placement, the sensors can be mounted on the outer linear actuators of each leg so that collisions or interference with other leg's sensors can be excluded.

Six inclination sensors are mounted on the six linear actuators of the hexapod. With these sensors, the position of the feet in relation to the hip can be determined. In addition, this sensor concept even provides information about the position and orientation of the robot in space.

In order to elude the direct kinematics problem, further concepts were studied in [21]. In these concepts, the unique solution of the direct kinematics problem can be obtained by using only the sensor data. This is generally achieved by means of additional sensors. Five different sensor concepts were proposed and presented in detail, which are described in the following.

Five angle sensors and one displacement sensor: By attaching one angle sensor on the linear actuator, two angle sensors on the lower joint, three angle sensors on the upper joint, and one displacement sensor on the linear actuator, the pose of this linear actuator is clearly determined just like the pose of the end-effector. The disadvantage of this concept is that all the sensors have to be placed on just one linear actuator. This might lead to difficulties in mounting the angle sensors on the joints or to limitations of the workspace.

Five angle sensors and one displacement sensor on an additional leg: An additional leg can be integrated in the middle of the hexapod. The leg can be made of a non-rotatable telescopic cylinder mounted between the upper and the lower platform. The length of the additional leg can be measured by the displacement sensor. The advantage of this concept is the possibility to avoid sensors on the linear actuators. In this context, angle sensors can be integrated much easier in the additional leg than in the joints of the linear actuators.

Four angle sensors and three displacement sensors: In this concept, two angle sensors and one displacement sensor are mounted on two linear actuators of the hexapod. Therewith, two vectors of known lengths and orientations are defined. The endpoints lie both on a plane indicating the orientation of the foot. Here, a straight line between these two points can be drawn. However, the rotation around the longitudinal axes of these two linear actuators is still undetectable. With an additional displacement sensor on another linear actuator, the pose of the foot can be obtained.

Four angle sensors, two displacement sensors, and three one-dimensional inclination sensors: Similar to the previous concept, two linear actuators are each equipped with a displacement sensor and two angle sensors at the universal joints. Therewith, again, a line on the foot platform is known. In this context, the plane through this line can only rotate around this line. The only remaining degree of freedom can be detected by means of an inclination sensor on the hip. However, this is only possible if the orientation of the foot platform with respect to the global coordinate system is known. Since this is not always the case, another inclination sensor on the foot is necessary to obtain meaningful data for calculating the hexapod's pose. Comparing the inclination of the hip and the foot platform, an angle between the two planes can be determined. The inclination sensors must be positioned so that they are mounted on the hip and the foot each in a perpendicular direction to the connecting line between the two linear actuators equipped with the other sensors.

Six displacement sensors and two angle sensors: In a concept with six displacement sensors, where each sensor is mounted on one linear actuator, direct kinematics offers up to 40 different poses for the robot [22]. With two additional angle sensors on two of the linear actuators, wrong poses can be excluded until only the correct solution remains.

4.3.2. Load Detection

Static stability can only be guaranteed if the ground projection of the robot's center of mass constantly remains in the stability region spanned by the C-shaped feet of the CENTAUROB, see Fig. 2. This point can be calculated from the forces along the linear actuators. Therefore, strain gauge force sensors are installed at the end of every linear unit. This makes it possible to measure the load of every leg. Furthermore, by knowing the actual pose of the CENTAUROB, the ground projection of the center of mass can be calculated using static equilibrium equations. In a static situation, the robot's center of mass can be calculated by the weights and positions of every center of mass of the individual component.

5. Kinematic Model

In this section, we review the kinematics of the linear actuators of the CENTAUROB. The complete description of the kinematics can be found in [5].

5.1. Kinematics of the Free Rigid Body

The kinematics of a body describes its possible movement without considering the cause of this movement, that is, the forces acting on the body. The movement of a single, free rigid body can be described by two different coordinate systems, a body-fixed one and an inertial one. The coordinate systems of the CENTAUROB are illustrated in Fig. 6. Here, $\{0\}$ denotes the inertial frame and $\{i\}$, $i = 1, \dots, 4$, denote the body-fixed frames.

The position and orientation of a platform can be described by the following vector of minimal coordinates:

$$\mathbf{y}_i = [p_{x,i} \ p_{y,i} \ p_{z,i} \ \alpha_i \ \beta_i \ \gamma_i]^\top = [\mathbf{y}_{t,i} \ \mathbf{y}_{r,i}]^\top, \quad i = 1, \dots, 4, \quad (1)$$

where $\mathbf{y}_{t,i}$ contains the translatory and $\mathbf{y}_{r,i}$ the rotatory degrees of freedom of the platform. Furthermore, $p_{x,i}$ shows in the forward, $p_{y,i}$ in the sideward, and

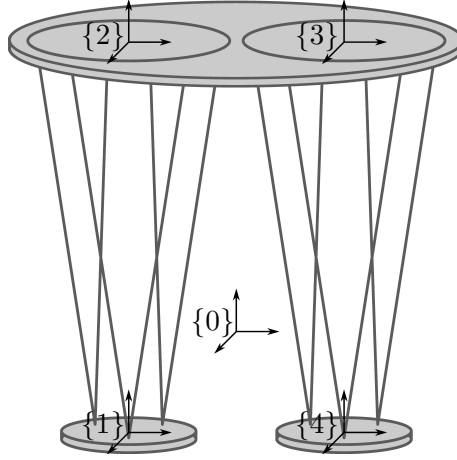


Figure 6. Coordinate systems of the CENTAUROB.

$p_{z,i}$ in the upper direction of the robot. The rotation angles α_i , β_i , and γ_i can be defined, for example, by CARDAN or EULER angles.

On the other hand, we can also use the lengths of the linear actuators $q_{k,i}$, $k = 1, \dots, 6$, for defining the minimal coordinates:

$$\mathbf{q}_i = [q_{1,i} \ q_{2,i} \ q_{3,i} \ q_{4,i} \ q_{5,i} \ q_{6,i}]^\top, \quad i = 1, \dots, 4. \quad (2)$$

In order to transform an arbitrary vector ${}^i\mathbf{p}$ described in the body-fixed frame $\{i\}$, $i = 1, \dots, 4$, into the inertial frame $\{0\}$, we use the following relation:

$$\mathbf{p} = \mathbf{p}_i + \mathbf{R}_i \cdot {}^i\mathbf{p}, \quad (3)$$

where $\mathbf{p}_i$ denotes the vector connecting the origins of frame $\{0\}$ and frame $\{i\}$ and $\mathbf{R}_i$ the rotation matrix from frame $\{i\}$ into frame $\{0\}$. Note that, here,

$$\mathbf{p}_i = \mathbf{y}_{t,i}. \quad (4)$$

Because the rotation angles in Eq. (1) do not span an orthogonal coordinate system, their temporal derivatives do not represent a geometrically meaningful quantity. For this reason, we have to introduce a new vector of minimal coordinates which only contains velocities:

$$\dot{\boldsymbol{\theta}}_i = [\dot{p}_{x,i} \ \dot{p}_{y,i} \ \dot{p}_{z,i} \ \omega_{x,i} \ \omega_{y,i} \ \omega_{z,i}]^\top = [\dot{\mathbf{y}}_{t,i} \ \boldsymbol{\omega}_i]^\top, \quad i = 1, \dots, 4. \quad (5)$$

Here, $\omega_{x,i}$, $\omega_{y,i}$, and $\omega_{z,i}$ denote the angular velocities around the the x , y , and z direction, respectively. Note that

$$\omega_i \neq \dot{\mathbf{y}}_{r,i} . \quad (6)$$

However, between ω_i and $\dot{\mathbf{y}}_{r,i}$, the following relationship holds:

$$\omega_i = \begin{bmatrix} \omega_{x,i} \\ \omega_{y,i} \\ \omega_{z,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \sin \beta_i \\ 0 & \cos \alpha_i & -\sin \alpha_i \cos \beta_i \\ 0 & \sin \alpha_i & \cos \alpha_i \cos \beta_i \end{bmatrix} \cdot \begin{bmatrix} \dot{\alpha}_i \\ \dot{\beta}_i \\ \dot{\gamma}_i \end{bmatrix} = \dot{\mathbf{y}}_{r,i} , \quad (7)$$

assuming that α_i , β_i , and γ_i are defined by CARDAN angles.

5.2. Inverse Kinematics of the CENTAUBOB

The inverse kinematics describes the relationship between the actuator coordinates of the active bodies $\mathbf{q}_i$, $i = 1, \dots, 4$, and the world coordinates $\mathbf{y}_i$, as illustrated in Fig. 7. For a hexapod, the world coordinates $\mathbf{y}_i$ cannot be computed directly. For this reason, we use the actuator coordinates $\mathbf{q}_i$ as a reference signal in our control system.

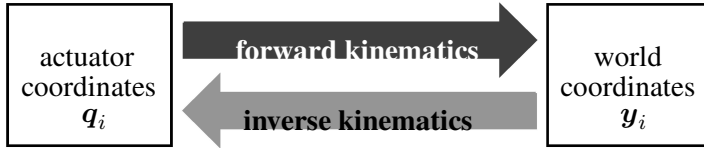


Figure 7. Relationship between the world and the actuator coordinates.

According to Fig. 8, the length of the k th linear actuator

$$q_{k,1} = q_{k,2} = |\mathbf{p}_{J_{k,2}J_{k,1}}| = |\mathbf{p}_{J_{k,1}J_{k,2}}| , \quad k = 1, \dots, 6 , \quad (8)$$

can be determined from

$${}^2\mathbf{p}_{J_{k,2}J_{k,1}} = {}^2\mathbf{p}_{21} + {}^2\mathbf{R}_1 \cdot {}^1\mathbf{p}_{1J_{k,1}} - {}^2\mathbf{p}_{2J_{k,2}} \quad (9)$$

with respect to the temporarily fixed frame $\{2\}$. Here, ${}^2\mathbf{R}_1$ denotes the rotation matrix from frame $\{1\}$ into frame $\{2\}$.

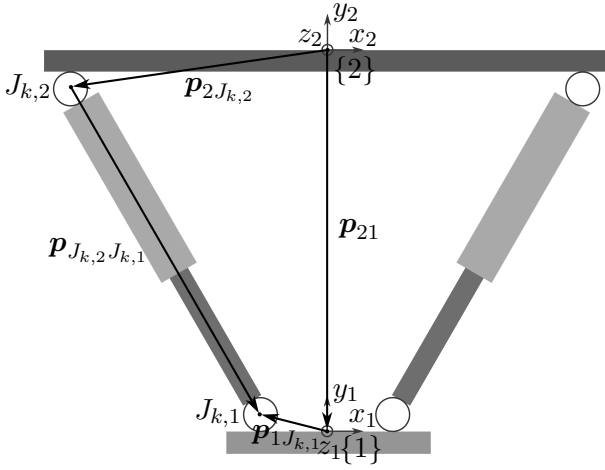


Figure 8. Schematic diagram for determining the length of a linear actuator.

For a given velocity vector $\dot{\theta}_i$, $i = 1, \dots, 4$, the velocities of the linear actuators $\dot{q}_i$ can be always determined by means of the inverse (geometrical) JACOBI matrix $\mathbf{J}_i^{-1}$:

$$\frac{d\mathbf{q}_i}{dt} = \mathbf{J}_i^{-1} \cdot \frac{d\theta_i}{dt}. \quad (10)$$

Because of the modular design of the CENTAUROB, only four structurally different inverse JACOBI matrices are needed:

- inverse JACOBI matrix of the linear actuators,
- inverse JACOBI matrix of the joints,
- inverse JACOBI matrix of the upper cylinder units,
- inverse JACOBI matrix of the lower cylinders.

In the following, we will review the inverse JACOBI matrix of the linear actuators. For the derivation of the other inverse JACOBI matrices, the reader is referred to [5].

5.3. Inverse JACOBI Matrix of the Linear Actuators

The velocities of the linear actuators $\dot{q}_{k,1}$, $k = 1, \dots, 6$, can be determined from

$$\dot{q}_{k,1} = \frac{d}{dt} |\mathbf{p}_{J_{k,2}J_{k,1}}| = \frac{1}{|\mathbf{p}_{J_{k,2}J_{k,1}}|} \cdot \mathbf{p}_{J_{k,2}J_{k,1}}^\top \cdot \dot{\mathbf{p}}_{J_{k,2}J_{k,1}}. \quad (11)$$

By using the relation

$$\dot{\mathbf{R}}_i \cdot \mathbf{R}_i^\top = \tilde{\boldsymbol{\omega}}_i = \begin{bmatrix} 0 & -\omega_{z,i} & \omega_{y,i} \\ \omega_{z,i} & 0 & -\omega_{x,i} \\ -\omega_{y,i} & \omega_{x,i} & 0 \end{bmatrix}, \quad (12)$$

the temporal derivative $\dot{\mathbf{p}}_{J_{k,2}J_{k,1}}$ can be calculated from

$$\dot{\mathbf{p}}_{J_{k,2}J_{k,1}} = \dot{\mathbf{p}}_{21} + \tilde{\boldsymbol{\omega}}_1 \cdot \mathbf{p}_{1J_{k,1}}. \quad (13)$$

Substituting in Eq. (11) and simplifying results in

$$\begin{aligned} \dot{q}_{k,1} = \frac{1}{|\mathbf{p}_{J_{k,2}J_{k,1}}|} \cdot & \left[\left(\mathbf{p}_{21} + \mathbf{p}_{1J_{k,1}} - \mathbf{p}_{2J_{k,2}} \right)^\top \cdot \dot{\mathbf{p}}_{21} \right. \\ & \left. + \left(\tilde{\mathbf{p}}_{1J_{k,1}} \cdot \left(\mathbf{p}_{21} - \mathbf{p}_{2J_{k,2}} \right) \right)^\top \cdot \boldsymbol{\omega}_1 \right]. \end{aligned} \quad (14)$$

Finally, the inverse JACOBI matrix of the linear actuators $\mathbf{J}_{A_1}^{-1}$ can be computed from the velocities of all six linear actuators:

$$\underbrace{\begin{bmatrix} \dot{q}_{1,1} \\ \vdots \\ \dot{q}_{6,1} \end{bmatrix}}_{\dot{\mathbf{q}}_1} = \underbrace{\begin{bmatrix} \dot{j}_{1,1} & \dot{j}_{1,2} \\ \vdots & \vdots \\ \dot{j}_{6,1} & \dot{j}_{6,2} \end{bmatrix}}_{\mathbf{J}_{A_1}^{-1}} \cdot \underbrace{\begin{bmatrix} \dot{\mathbf{p}}_{21} \\ \boldsymbol{\omega}_1 \end{bmatrix}}_{\dot{\boldsymbol{\theta}}_1}, \quad (15)$$

where

$$\dot{j}_{k,1} = \frac{\left(\mathbf{p}_{21} + \mathbf{p}_{1J_{k,1}} - \mathbf{p}_{2J_{k,2}} \right)^\top}{|\mathbf{p}_{J_{k,2}J_{k,1}}|}, \quad (16)$$

$$\dot{j}_{k,2} = \frac{\left(\tilde{\mathbf{p}}_{1J_{k,1}} \cdot \left(\mathbf{p}_{21} - \mathbf{p}_{2J_{k,2}} \right) \right)^\top}{|\mathbf{p}_{J_{k,2}J_{k,1}}|}. \quad (17)$$

6. Control Concepts

The motion control strategy for the CENTAUROB consists of three parts: the execution of user commands or wishes, the stability control, and the position control. Although the core of the strategy is formed by the execution of user commands or wishes, the parts are structured as a three-step cascade. The stability control is designed to ensure static stability at all times while the position control should make sure that the target position is always reached. All parts work independently of each other by using different control parameters as well as query and intervention times.

6.1. Execution of User Commands

Generally, the movement of the CENTAUROB can be classified into statically stable and dynamic movement. Here, we will concentrate on the statically stable movement. Furthermore, it has to be distinguished between a single-step and a multi-step movement. In the single-step movement, the robot makes only one step, whereas, in the multi-step movement, the CENTAUROB can reach a target position out of his working area by making several steps. This requires a more complex program than in the single-step movement. However, both movements are nearly the same except that in the multi-step movement, the CENTAUROB still has a command in mind after one step was made.

The overall user command execution concept is structured as follows. After a command is entered, it is checked for its sense. Furthermore, it is necessary to check if the parameters (these are, for example, the maximum and minimum velocity, the acceleration, and the step size) remain in their boundaries. For this reason, the robot's pose must be known exactly. The step size, for example, depends on the height of the hip and the step height, as illustrated in Fig. 9. We can see that the maximum step size of 680 mm can be reached at a hip height of 937 mm. In the retracted or extended hip condition, no step size can be approached.

The verification of the user command is followed by the calculation of the motion trajectories for every part of the motion by using the inverse kinematic model from Section 5. The calculated trajectories are used to simulate the motion and to check the parameters one more time. After the trajectory is successfully checked, the trajectories are enabled and sent to the motors. After the user command is executed, the CENTAUROB returns to the initial position,

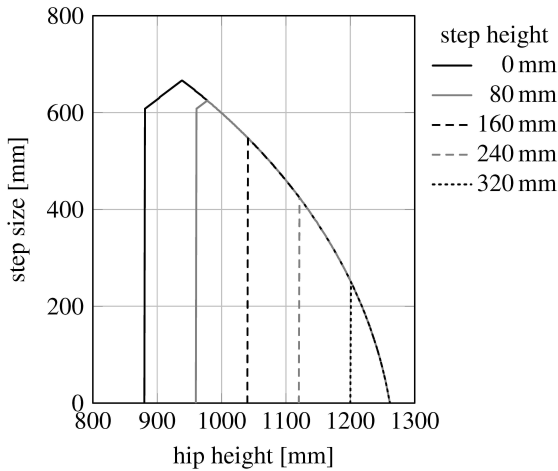


Figure 9. Step size of the CENTAUROB in longitudinal direction as a function of the hip height and the step height.

phase zero, and the program is waiting for the next command. The overall user command execution concept can be viewed as a loop starting with an entered command and returning to the initial position after executing this command, see Fig. 10.

6.2. Stability Control Concept

Static walking can be realized if the acceleration of the hip and the resulting forces of the moving bodies can be neglected. This is possible if the weight of the hip is significantly higher than the weight of each leg and the movement of the hip is relatively slow. Static stability can then be reached if the ground projection of the robot's center of mass constantly remains in the stability region, see Fig. 2. It can be calculated from the data of the force sensors as described in Section 4.3.2.

6.3. Position Control Concept

The position control concept is used to make sure that the CENTAUROB reaches his target position. This can be realized, for example, by a very long interference time. This means that the execution of user commands and the sta-

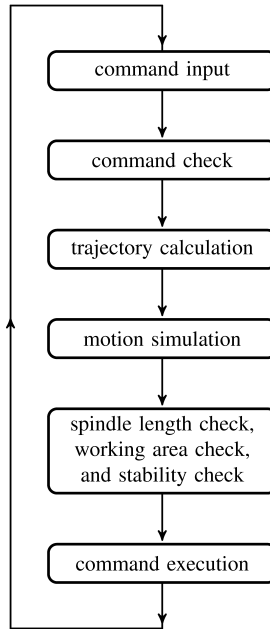


Figure 10. Overall user command execution concept for the motion control.

bility control act faster than the position control. Therewith, it is ensured that stability is given and the user commands are executed properly. This concept leads to long follow-up times until the target position is finally reached. Furthermore, large differences between the actual and the target position are possible, which could cause collisions.

6.4. Power Control Concept

On top of the hip, two rechargeable batteries with 24 V DC are installed to supply the electrical components. These are the servo motors supplied by their motor cards and the breaks that are needed to hold the motor's position in case of an electricity failure. The batteries use lithium anodes and nickel-manganese-cobalt cathodes which leads to a very high specific energy. In order to have two independent legs with their own power supply, it is useful to separate the power supply of each leg. Because the non-constant load on the feet during one step results in a non-constant power consumption, it is necessary to analyze the

consumed current during one step.

In this context, three kinds of foot loads must be distinguished that appear in the following phases: the double support phase, the single support phase, and the swing phase. In the double support phase, both feet stand still on the ground, and the load of the hip is distributed more or less symmetrically on the feet, depending on the position of the hip and the ground conditions. In this configuration, each leg requires nearly the same current, which can be easily calculated for static conditions. Here, the whole load is concentrated on only one leg because the other leg is in the so-called swing phase. In the swing phase, one leg is moved to its target position, so the motors of this leg only have to carry the weight of the corresponding foot. In this phase, the leg is stressed with a tensile load whereas in the other phases, the leg is stressed with a compressive load.

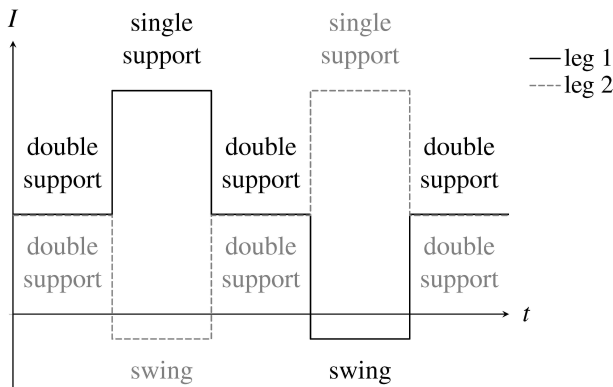


Figure 11. Overall current consumption I over time t during a walking step.

The three phases alternate during the motion process. The overall current consumption during a walking step is shown in Fig. 11. In order to reduce the average current that lays between the currents of the double support phase and the single support phase, the holding breaks can be used during the single support phase. Another strategy to reduce the average current is the optimization of the walking pattern and trajectory by the parameters step size, step height, step velocity, and acceleration.

7. Walking Scenario Simulations

In order to prove the agility of a robot with two serially arranged parallel kinematics, several walking scenarios were simulated. Although the robot's focus is walking forwards, it is still possible to walk sideways, climbing straight and spiral stairs, overcome obstacles, turning around, and some more special features. The workspace boundaries are illustrated in Fig. 12.

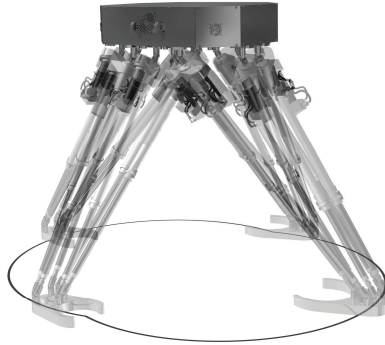


Figure 12. Visualization of the robot's workspace boundaries: ± 680 mm in longitudinal and lateral direction with respect to the center points of the hexapods.

7.1. Walking Pattern

The statically stable movement of the CENTAUROB can be divided into six phases, where phase zero builds the so-called initial situation. In this initial situation, the CENTAUROB has finished the last movement or is freshly started, and the program is waiting for a user command. The phases one to five are part of every movement of the CENTAUROB. The only parameters that change are the hip height, the step length and height, the speed, and the orientation of the feet. Starting from the initial position where the CENTAUROB stands still with both feet central under his hip (see Fig. 5), in phase one, the hip is moved over the bigger foot to increase the stability and to relieve the other foot, see Fig. 13(a).

In phase two, the released foot moves along the calculated trajectory towards the target position, where the hip and the other foot remain still, see Figs. 13(a)–(b). This trajectory is quite complex because in the initial position of the feet, the

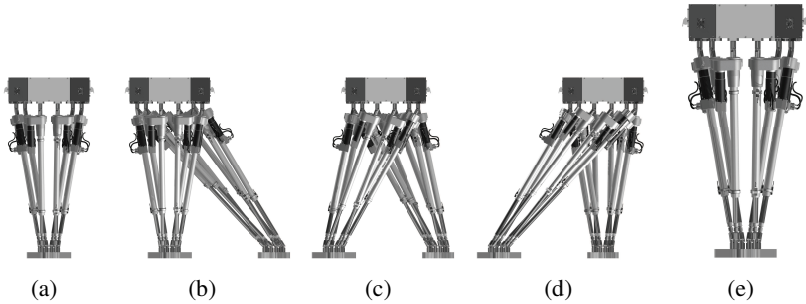


Figure 13. Motion states of the CENTAUBOB.

bigger foot nearly surrounds the smaller foot. There are three different possible trajectories for reaching the target position without any foot collisions. One possibility is to lift the smaller foot to the height of the bigger foot and then to start the trajectory. The second possibility is to move the smaller foot sideways out of the bigger foot and then to start the trajectory. Finally, it is also possible to move one of the feet with a complex trajectory that prevents all collisions by precalculation. Therefore, the actual position and the size of the feet have to be known exactly.

In phase three, the hip moves from the position above the bigger foot to the one above the smaller foot, see Figs. 13(b)–(d). Depending on the weight of the hip, it might not be necessary to place the hip directly above the smaller foot. This would make it possible to move the hip straight in the direction of the target without any lateral movement.

In phase four, the bigger foot is moved under the hip, see Figs. 13(d)–(e). As mentioned in phase two, a complex trajectory is useful to prevent from any foot collision. Therefore, a fifth phase is necessary to move the hip into the initial position in the middle of the feet if a hip movement above the stable standing foot was necessary in the former phases, see Fig. 13(e).

In order to achieve an 'economic' walking, several modifications on the walking pattern described above were made. In this context, 'economic' means to use an optimized force distribution on the linear actuators to reduce the current consumption of the batteries and increase the walking range. Furthermore, 'economic' stands for a walking pattern that is definite, easily adjustable, with low complexity, and identical for both feet.

In phase one and five, a hip movement between zero and 160 mm is required

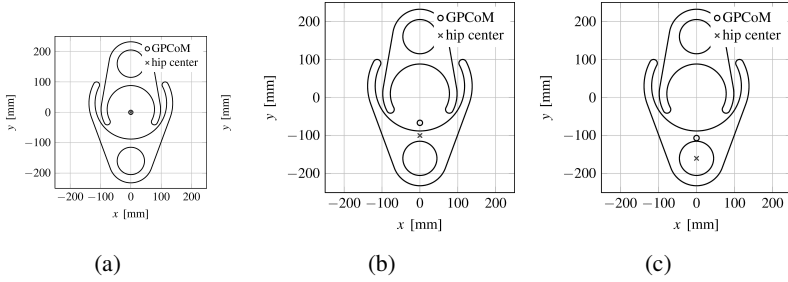


Figure 14. Ground projection of the robot's center of mass (GPCoM) at hip displacements of (a) 0 mm, (b) 100 mm, and (c) 160 mm.

to stabilize the motion process. This hip movement over the bigger foot reduces the forces along the linear actuators in the supporting leg. The ideal amount of hip movement depends on the weight of the hip because it affects the forces along the linear actuators as well as the lever arm of the tipping torque by the height of the center of mass. For example, when moving the hip for 100 mm, the forces along the linear actuators are reduced to 55% of the forces with no hip movement. For a hip movement of about 160 mm, the forces along the linear actuators are even reduced to a third. Figure 14 shows the ground projection of the robot's center of mass for different hip displacements.

There are a few parameters influencing the foot trajectory. The step size, for example, has two limits, the minimum step size, which is 238 mm and results from the shape of the feet, and the maximum step size that depends on the step height, the hip movement during a walking step, and the height of the hip as shown in Fig. 9.

Phase three, where the hip is moved from the position above the bigger foot to the one above the smaller foot, is necessary to move the center of mass from the position above one foot to the position above the other foot. The trajectory is statically stable because in this so-called double support phase, the ground projection of the center of mass always remains in the stability region formed by both feet standing on the ground as illustrated in Fig. 15(a). Figure 15(b) shows the ground projection of the robot's center of mass for a whole walking step.

For huge step sizes, for example, 680 mm in longitudinal or 640 mm in lateral direction, in phases one and four, it gets necessary to move the hip contrary to the moving direction of the CENTAUROB. This ensures that the ground pro-

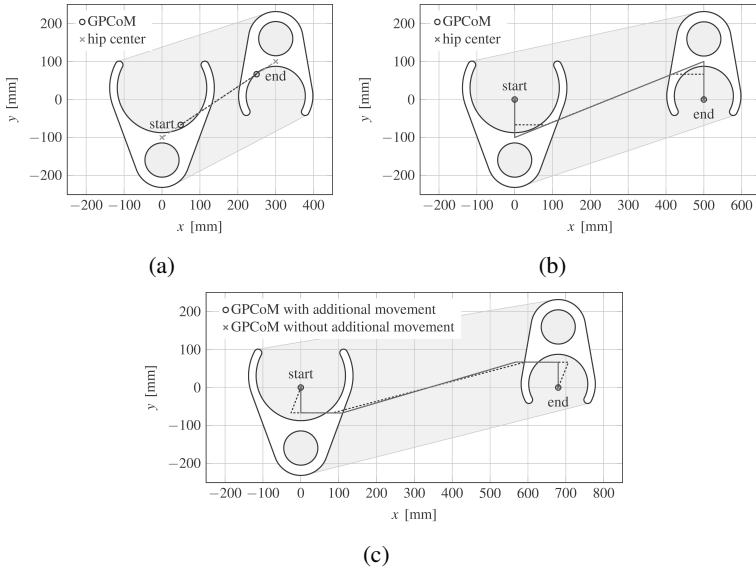


Figure 15. Ground projection of the robot's center of mass (GPCoM): (a) in phase three, (b) for a whole walking step, and (c) for huge walking steps with and without additional hip displacement.

jection of the center of mass remains in the stability region. Figure 15(c) shows the differences of the ground projection of the center of mass with and without an additional hip movement in the phases one and five for forward walking.

7.2. Walking Forwards and Sideways

The forward walking of the CENTAUROB is already described above. Additionally to this movement, where the second foot only follows the first one, it is also possible to perform human-like steps where one foot overtakes the other one. Both forward movements have their advantages. For example, the first scenario leads to lower forces in the linear actuators and therewith to a lower current consumption. Furthermore, it preserves a higher amount of stability since the displacement of the ground projection of the center of mass in the walking direction in phase two depends on the step size. The second scenario is a more familiar motion. Due to the omission of an intermediate step, a higher moving velocity for the same step sizes and step velocities can be achieved. This second

moving scenario, however, has a disadvantage in terms of an economic walking pattern. After one step, the robot is not in its initial pose. In order to return to the initial, stable pose, another step is necessary. Both walking scenarios with its differences are illustrated in Figs. 16 and 17.

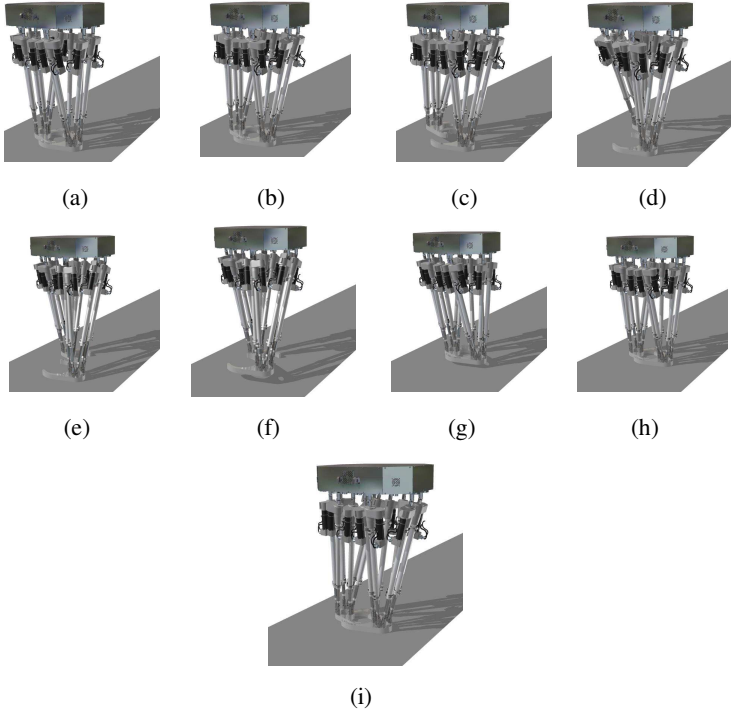


Figure 16. Moving forwards, sequence one: Moving the hip above the bigger foot (a)–(b), lifting the smaller foot (b)–(c), moving the smaller foot forwards to the target position (c)–(d), moving the hip above the smaller foot (d)–(e), lifting the bigger foot (e)–(f), moving the bigger foot forwards to the position next to the smaller foot (f)–(g), moving the hip to the initial position above the middle of the feet (g)–(h), starting the next step again with the hip movement (h)–(i).

In order to avoid obstacles, it might be useful to walk sideways instead of turning around and walking forwards. The sideward movement, shown in Fig. 18, mainly equals the forward movement. The main difference between these movements in the walking pattern is the larger hip movement that is nec-

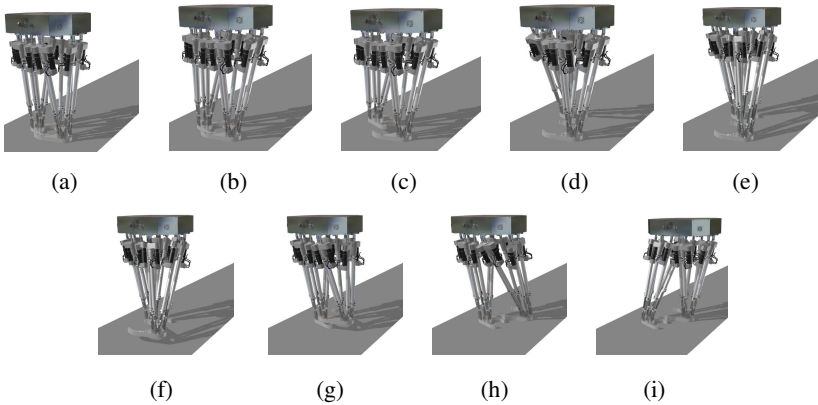


Figure 17. Moving forwards, sequence two: Moving the hip above the bigger foot (a)–(b), lifting the smaller foot (b)–(c), moving the smaller foot forwards to the target position (c)–(d), moving the hip above the smaller foot (d)–(e), lifting the bigger foot (e)–(f), moving the bigger foot forwards to the target position in front of the smaller foot (f)–(h), moving the hip above the bigger foot (h)–(i).

essary to achieve stability during walking. This hip movement is contrary to the moving direction.

7.3. Special Features

Due to the six degrees of freedom of the hip and each foot, the robot provides plenty of special moving features. In case an obstacle shows up or it becomes necessary to climb steps or stairs, the robot can change the height of the hip as well as the step height. Therewith, it is not only possible to overcome small obstacles with heights up to 350 mm and lengths up to 300 mm, as shown in Fig. 19, the robot can also climb steps and stairs with rises up to 400 mm, as shown in Fig. 20.

For statically stable walking, two cases for overcoming an obstacle are possible. In the first case, the obstacle is small enough to overcome it within one step. Here, no additional or only little hip movement is necessary so that the ground projection of the robot's center of mass remains in the stability region. In the second case, the obstacle has an even surface to place a foot on top of the obstacle. Here, the obstacle is passed within two or more steps. It is also possible to overcome bigger obstacles but not for statically stable walking.

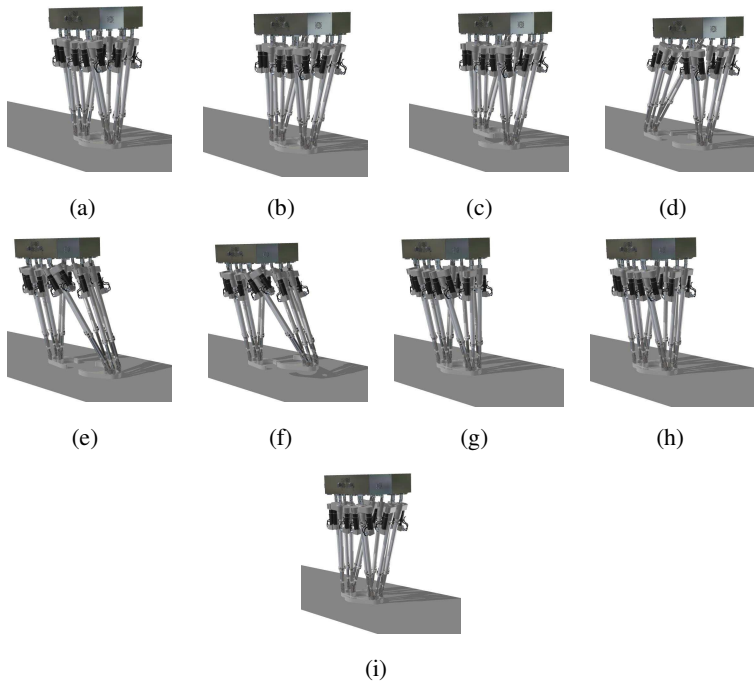


Figure 18. Moving sideways: Moving the hip above the bigger foot (a)–(b), lifting the smaller foot (b)–(c), moving the smaller foot sideways to the target position (c)–(d), moving the hip above the smaller foot (d)–(e), lifting the bigger foot (e)–(f), moving the bigger foot sideways to the position next to the smaller foot (f)–(g), moving the hip to the initial position above the middle of the feet (g)–(h), starting the next step again with the hip movement (h)–(i).

In order to operate in any environment, it is essential not only being able to walk a curve but also being able to turn around. For example, in tight hallways, there is not enough space to turn in a curve. Due to the specially C-shaped feet of the CENTAUROB, it is possible to turn around nearly at a point and with only a few steps. Figure 21 shows the scenario for the robot rotating by 90 degrees.

By combining the two motions, climbing stairs and turning around, the robot is able to walk spiral staircases as shown in Fig. 22. This is only possible because of the special C-shaped feet. For the shown movement, the center of the hip never leaves the stability region formed by the feet. This motion makes the CENTAUROB a unique moving device with several possible applications as

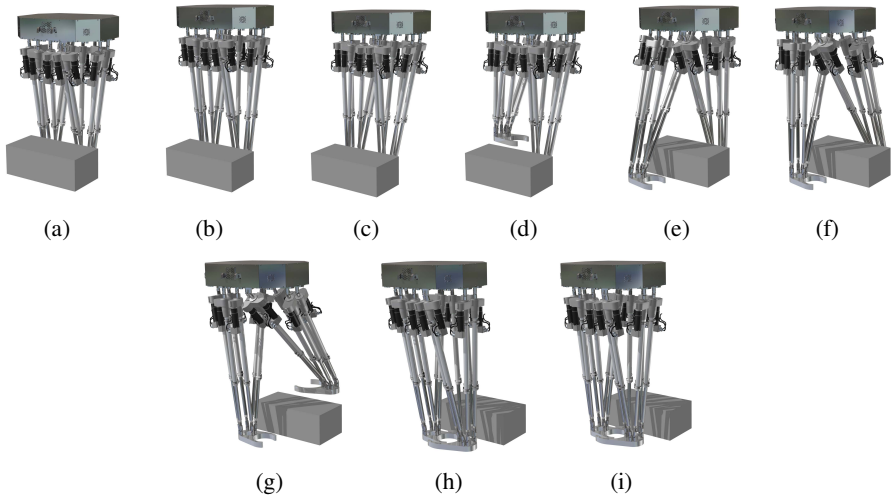


Figure 19. Overcoming obstacles: Lifting the hip to the designated height (a)–(b), moving the hip above the bigger foot (b)–(c), lifting the smaller foot above the height of the obstacle (c)–(d), moving the smaller foot above the obstacle to the target position (d)–(e), moving the hip above the smaller foot (e)–(f), lifting the bigger foot above the height of the step (f)–(g), moving the bigger foot to the position next to the smaller foot (g)–(h), moving the hip to the initial position above the middle of the feet (h)–(i).

will be discussed in the next section.

8. Fields of Usage and Applications

As a universal moving platform, the CENTAUROB can be used in several applications and environments. One possible field of usage is, for example, the industry. An assisting or fully autonomous employment of robots in unpaved terrain will play an important role in the future. Despite of the working area, a lot of jobs will be transferred to robots due to their precision, velocity, safety, and endurance. Here, parallel two-legged robots can be used, for example, as industrial carrying devices in undefined environments or even as autonomous robots in rough or dangerous areas.

Another aspect, that must be thoughtfully treated, is the increasing average

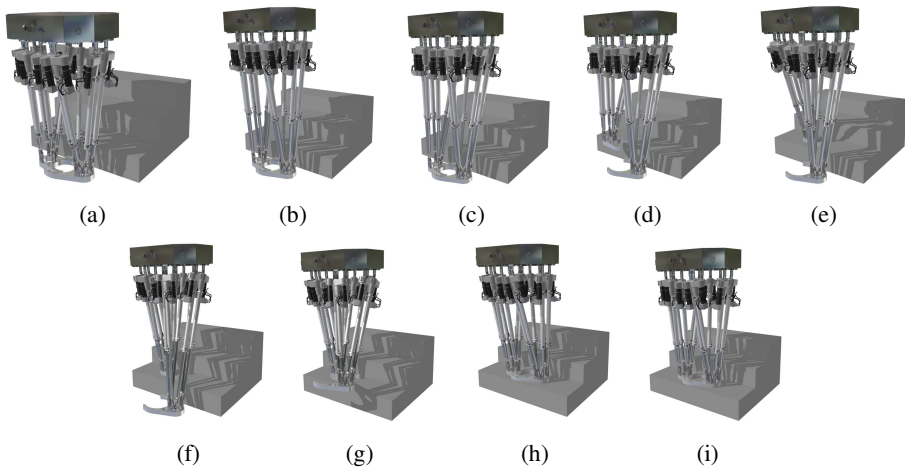


Figure 20. Climbing straight stairs: Lifting the hip to the designated height (a)–(b), moving the hip above the bigger foot (b)–(c), lifting the smaller foot above the height of the step (c)–(d), moving the smaller foot to the target position (d)–(e), moving the hip above the smaller foot (e)–(f), lifting the bigger foot above the height of the step (f)–(g), moving the bigger foot to the position next to the smaller foot (g)–(h), moving the hip to the initial position above the middle of the feet (h)–(i).

age of our society. Despite of this fact, the aim to preserve a self-determined life with coping daily routines becomes stronger. However, for older people, after a certain point in time, assistance and/or care become inevitably necessary. In this context, a survey in Germany found out that the majority of the German citizens does not want to move into a retirement home in old age [23]. Two-thirds of the respondents want to live in an apartment or a house when they are old and only 15% prefer a nursing or retirement home. Nowadays, mobility and certain independence have a high individual and social value. Furthermore, they are an important aspect of freedom and self-determination, that is, dignity.

One possibility to retain the mobility of elderly people is by means of technical support. Wheelchairs and rollators usually require flat environments without fixed or exposed obstacles. Fixed obstacles are, for example, door thresholds in the apartment or terrace entries. Making things worse, wheelchairs and rollators have often to be lifted or carried to get, for example, in the apartment or

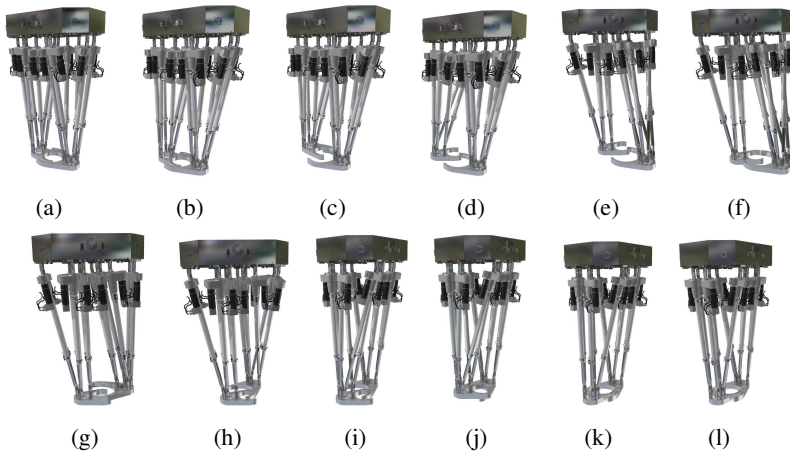


Figure 21. Rotating around 90 degrees: Moving the hip above the bigger foot (a)–(b), lifting the smaller foot (b)–(c), turning the smaller foot (c)–(d), rotating the hip and moving it above the smaller foot (d)–(e), lifting the bigger foot (e)–(f), rotating and moving the bigger foot to the target position (f)–(h), rotating the hip and moving it above the smaller foot (h)–(i), lifting the smaller foot (i)–(j), turning the smaller foot (j)–(k), moving the hip to the initial position above the middle of the feet (k)–(l).

in a public transportation vehicle. In case of using a stair lift, the person even has to change seats which might be a huge effort or even not be possible at all. However, so far, existing system improvements have not shown any significant progress in overcoming these issues. For this reason, using wheels as locomotion concept for elderly people might be reconsidered on behalf of using a two-legged walking robot.

Conclusion

In this chapter, we reviewed the parallel two-legged walking robot named CENTAUROB, a novel robot system for statically stable walking. Due to the universal moving platform, the robot offers plenty of possible applications, for example, as a sustainable mobility assurance in the assistance and care or as an assisting or fully autonomous industrial robot in unstructured or dangerous environments. By using two serially arranged parallel kinematics (hexapods) as

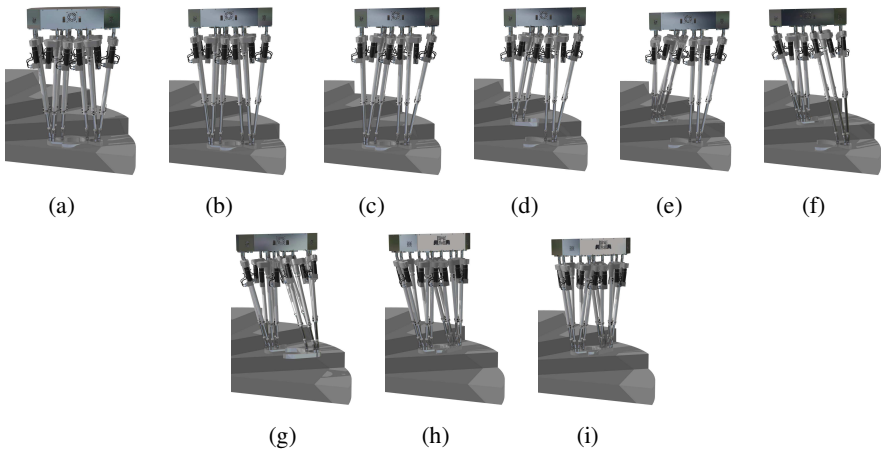


Figure 22. Climbing spiral stairs: Lifting the hip to the designated height (a)–(b), moving the hip above the bigger foot (b)–(c), lifting the smaller foot above the height of the step (c)–(d), turning the smaller foot and moving it to the target position (d)–(e), turning the hip and moving it above the smaller foot (e)–(f), lifting the bigger foot above the height of the step (f)–(g), turning the bigger foot and moving it to the position next to the smaller foot (g)–(h), turning the hip and moving it to the initial position above the middle of the feet (h)–(i).

leg structures, this system totally renounces the use of wheels as the locomotion concept which provides high mobility in every direction and enables walking in unpaved terrain as well as the handling of steps, stairs, and low clearances. In order to prove the agility of the universal robot, several walking scenarios such as moving forwards, backwards, and sideways as well as other motions were presented.

We also described the generation of an economic walking pattern for statically stable walking of the CENTAUBOB. In order to minimize the calculation effort, the movement of the robot is separated into as few segments as possible. This is why for a whole step of the robot, only five different trajectories are needed, whereby the trajectories for both feet are identical. Furthermore, we described the necessary hip movement to reduce the forces among the linear actuators and to improve the stability.

In order to detect the actual pose of the CENTAUBOB and its orientation relative to the ground, several sensor concepts were presented and discussed.

Knowing the pose of the robot, it is possible to focus on the localization and path planning in an unknown environment. An algorithm for robot path planning and obstacle avoidance was already developed at our workgroup [24].

At the current project level, we consider a walking pattern based on feed forward control using precalculated steps. In the next level, the development of a much more sophisticated closed-loop control system is in the focus accompanied by a significantly extended sensor structure that shall provide the position, acceleration, as well as visual information. The aim is to establish a real-time control that allows an online computation of the entire robot motion with respect to its environment.

References

- [1] Hirose, M., and Ogawa, K., 2007. "Honda humanoid robots development". *Philosophical Transactions of the Royal Society A*, **365**, pp. 11–19.
- [2] Lohmeier, S., Buschmann, T., and Ulbrich, H., 2009. "Humanoid robot LOLA". In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 775–780.
- [3] Lim, H., and Takanishi, A., 2007. "Biped walking robots created at Waseda University: WL and WABIAN family". *Philosophical Transactions of the Royal Society A*, **365**, pp. 49–64.
- [4] Schlattmann, J., 1997. "Development of a highly flexible two-legged walking device". In *Proceedings of the 4th International Conference on Computer Integrated Manufacturing*, pp. 1287–1294.
- [5] Du, S., Schlattmann, J., Schulz, S., and Seibel, A., 2015. "New prototype of the two legged robot CENTAUBOB". In *Proceedings of the ASME 2015 International Mechanical Engineering Congress & Exposition*, Paper ID 50354.
- [6] Schlattmann, J., and Hampel, J., 2000. "Performance of the CENTAUBOB at production plants". In *Proceedings of the 5th International Conference on Computer Integrated Manufacturing*, pp. 425–433.
- [7] Schlattmann, J., Tesfu, M. T., and Ziemer, L., 2007. "New steps in the development of the two legged robot CENTAUBOB". *The International Journal of Modern Engineering*, **7**(2).

- [8] Gough, V. E., and Whitehall, S. G., 1962. “Universal tire test machine”. In *Proceedings of the 9th International Technical Congress FISITA*, Vol. 117, pp. 117–137.
- [9] Stewart, D. A., 1966. “Platform with six degrees of freedom”. *Proceedings of the Institution of Mechanical Engineers*, **180**(15), pp. 371–386.
- [10] Clavel, R., 1988. “Delta, a fast robot with parallel geometry”. In *Proceedings of the 18th International Symposium on Industrial Robots*, pp. 91–100.
- [11] Garcia, E., Jimenez, M. A., Santos, P. G. D., and Armada, M., 2007. “The evolution of robotics research”. *IEEE Robotics & Automation Magazine*, **14**(1), pp. 90–103.
- [12] Evans, J. M., 1994. “Helpmate: an autonomous mobile robot courier for hospitals”. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1695–1700.
- [13] Baalbaki, H., Lamiri, M., and Xie, X., 2008. “Joint location and configuration of mobile service robots for hospitals”. In *Proceedings of the IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 615–620.
- [14] Ozkil, A., Fan, Z., Dawids, S., Aanes, H., Kristensen, J., and Christensen, K., 2009. “Service robots for hospitals: A case study of transportation tasks in a hospital”. In *Proceedings of the IEEE International Conference on Automation and Logistics (ICAL)*, pp. 289–294.
- [15] Riebe, S., and Ulbrich, H., 2003. “Modelling and online computation of the dynamics of a parallel kinematic with six degrees-of-freedom”. *Archive of Applied Mechanics*, **72**(11–12), pp. 817–829.
- [16] Du, S., Schlattmann, J., Schulz, S., and Seibel, A., 2016. “Passive rotation compensation in parallel kinematics using quaternions”. *Proceedings in Applied Mathematics and Mechanics*, **16**. To appear.
- [17] Arai, T., Cleary, K., Nakamura, T., Adachi, H., and Homma, K., 1990. “Design, analysis and construction of a prototype parallel link manipulator”. In *Proceedings of the IEEE International Workshop on Intelligent Robots and Systems (IROS)*, pp. 205–212.

-
- [18] Stoughton, R., and Arai, T., 1991. "Optimal sensor placement for forward kinematics evaluation of a 6-dof parallel link manipulator". In Proceedings of the IEEE/RSJ International Workshop on Intelligent Robots and Systems (IROS), pp. 785–790.
 - [19] Merlet, J.-P., 1993. "Closed-form resolution of the direct kinematics of parallel manipulators using extra sensors data". In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp. 200–204.
 - [20] Riebe, S., 2005. *Aktive Schwingungsisolierung und Bahnregelung von Hexapodsystemen*. Fortschritt-Bereiche VDI, Düsseldorf, Germany.
 - [21] Tesfu, M. T., 2013. *Systematic Development of a Real-time Motion Control System for a Coupled Parallel Kinematics Robot*. Der Andere Verlag, Uelvesbuell, Germany.
 - [22] Raghavan, M., 1993. "The stewart platform of general geometry has 40 configurations". *Journal of Mechanical Design*, **115**(2), pp. 277–282.
 - [23] Welt, D., 2011. Deutsche wollen im Alter selbstständig wohnen, January. Online; posted 17-January-2011.
 - [24] Shehata, H., and Schlattmann, J., 2012. "Reactive algorithm for mobile robot path planning among moving target, obstacles by means of dynamic virtual obstacle concept". In Proceedings of the International Conference on Flexible Automation and Intelligent Manufacturing (FAIM).

Chapter 5

SLIDING-MODE TRACKING CONTROL OF THE 6-DOF 3-LEGGED WIDE-OPEN PARALLEL ROBOT

Mohammad H. Abedinnasab^{1,}, Jaime Gallardo-Alvarado²
Bahram Tarvirdizadeh³ and Farzam Farahmand^{4,5}*

¹Department of Biomedical Engineering, Rowan University,
Glassboro, New Jersey, US

²Department of Mechanical Engineering,
Instituto Tecnológico de Celaya, TNM,
Celaya, GTO, México

³Faculty of New Sciences and Technologies,
University of Tehran, Tehran, Iran

⁴School of Mechanical Engineering,
Sharif University of Technology, Tehran, Iran

⁵Research Center of Biomedical Technology and Robotics,
Tehran University of Medical Sciences, Tehran, Iran

Abstract

In this chapter, we analyze and robustly control the 6-DOF 3-legged Wide-Open parallel manipulator, using a Lyapunov analysis approach. The mechanism is evaluated in terms of kinematics, workspace, and dynamics. By implementing the dynamic model, tracking control of the

*E-mail address: abedin@rowan.edu

moving platform is performed using the nonlinear sliding-mode control method. The numerical simulations with desired trajectory inputs attest to the excellent performance and robustness of the designed position tracking sliding-mode controller, in presence of fast varying uncertain parameters and external disturbances, which are common in parallel manipulators.

1. Introduction

Parallel manipulators have many distinct advantages over their serial counterparts; such as greater stiffness, accuracy, speed, and payload capabilities. Those are mainly due to the parallel linkages which distribute the payload and average out the positioning errors, since without parallel chains, the payload and positioning errors accumulate [1, 2, 3, 4].

Many researchers have been studying on those manipulators after Gough and Whitehall [5] first introduced a 6-DOF parallel manipulator with an application in tire-testing equipment, followed by Stewart [6], who designed a similar robot to be used as a flight simulator, known as the Gough-Stewart platform. The Gough-Stewart platform consists of two platforms, one fixed and the other moving, connected with six extensible legs. Many variants of the Gough-Stewart platform have been introduced in the literature for different applications [1, 7, 8, 9, 10, 11, 12, 13, 14, 15].

However, the lower number of legs induce less interferences between the kinematic chains, enlarging the workspace of the robot. Monsarrat and Gosselin [16] designed a 3-legged 6-DOF mobile platform connected to the base by three identical kinematic chains using five-bar linkages. Badescu and Mavroidis [17] performed the workspace optimization of 3-legged translational UPU and orientational UPS parallel platforms under joint constraints. Lu et al. [18] proposed a 4-DOF over-constrained RRPV + 2UPU parallel manipulator (PM) with 3 legs. Zhang and Zhang [19] proposed a novel 2-DOF parallel manipulator with 3 legs, which is applied in vehicle driving simulators. Coppola et al. [20] designed a 3-legged self-reconfigurable architecture. It addresses the realm of reconfigurable 6-DOF parallel mechanisms, for sustainable manufacturing. Azulay et al. [21] proposed a 3-legged 6-DOF architecture based on a 3-PPRS topology that can achieve high (end-effector) tilt angles with enhanced stiffness.

The purpose of the present study is to analyze and control the 6-DOF 3-legged Wide-Open parallel robot for being used in various applications. The

proposed parallel manipulator is a 6-DOF spatial platform mechanism. It consists of a fixed base platform and a moving platform, which are connected by three extensible legs. The mechanism enjoys a unique non-symmetric architecture with a frontally open half-plane, enabling it to easily embrace and manipulate column-shape objects.

On the other hand, accurate trajectory tracking control of parallel robots is a key system requirement, as these devices must often follow prescribed motions, and high performance control strategies can significantly improve the tracking performance [22, 4]. Tracking control of parallel robots has been approached using both linear and nonlinear control laws [23, 4].

There are two design frameworks for the control scheme of the parallel robots. One is to design a controller based on the joint space coordinate and the other is based on the workspace coordinate. The former is a conventional control idea, which is used to design a controller to set the actuators' displacements based on the desired displacements computed from the position command of the manipulator by inverse kinematics. If the actuators' displacements are well controlled, the moving platform of the parallel manipulator moves based on the reference trajectory. Many controllers in the literature are based on the joint space coordinate [22, 24, 2]. However, the controllers based on the workspace coordinate are directly controlled based on the position of the moving platform, and not the actuators.

The rest of the chapter is organized as follows. In Section 2, the Wide-Open parallel mechanism is introduced and described. The architecture of the mechanism is analyzed and an experimental evaluation is performed. The kinematic analysis of the mechanism is performed in Section 3. Workspace analysis with considering different constraints on the spherical joints is presented in Section 4. In Section 5, inverse and forward dynamic analyses of the mechanism are discussed, and a numerical simulation is performed. Position control of the robot using the sliding-mode algorithm, as well as illustrative numerical simulations are reported in Section 6. Finally, a general conclusion on the chapter is provided in Section 7.

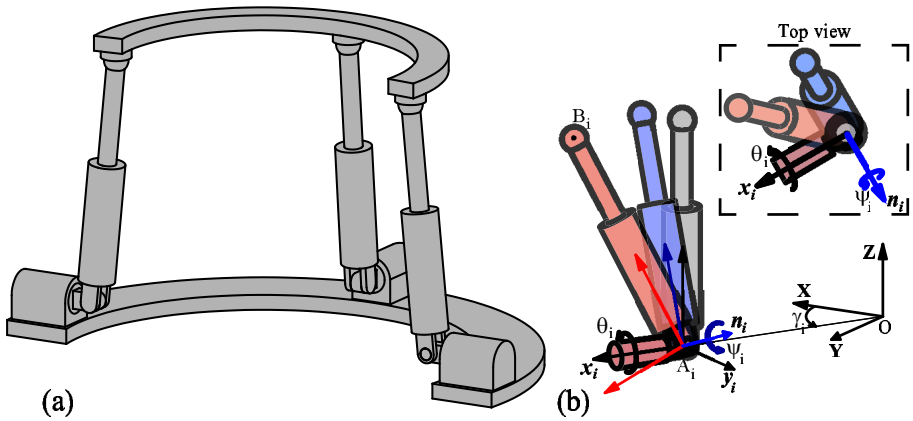


Figure 1. (a) Illustration of the 6-DOF 3-legged Wide-Open parallel manipulator. There are two actuators per leg. Rotary actuators are resting on the fixed platform, and linear actuators are moving. (b) Kinematic variables of i -th leg are shown. Active rotation is θ_i around x_i axis, along with the passive rotation of ψ_i around n_i axis. A linear actuator sets the length d_i , which is the distance between A_i and B_i .

2. The Wide-Open Robot Description

2.1. The Manipulator's Architecture

The schematics of the 6-DOF 3-legged Wide-Open mechanism is shown in Figure 1 (a). The mechanism is another version of the basic 3-legged parallel manipulator with symmetrical legs [25, 26, 27]. Each leg is composed of three joints; universal, prismatic, and spherical (Figure 1 (b)). A rotary and a linear actuator are used to actuate each leg. The rotary actuators, whose shafts are attached to the lower parts of the linear actuators through universal joints, are placed on a semicircle on the fixed platform. The spherical joints connect the upper parts of the linear actuators to the moving platform.

By replacing the passive universal joints in the Stewart mechanism with active joints in the proposed mechanism, the number of legs could be reduced from 6 to 3. This change makes the mechanism lighter, since the rotary actuators are resting on the fixed platform, which allows for higher accelerations to be available due to smaller inertial effects. More importantly, in the proposed

parallel manipulator, the legs are configured non-symmetrically on a semicircle on the base and moving platforms. This frontally wide open architecture, enables the mechanism to embrace and manipulate column-shape objects. The applications of such mechanism are versatile, from fracture reduction of long bones in surgical robotics to column climber robots in industrial robotics.

As shown in the Figure 1, coordinate $C_i(A_i, x_i, y_i, z_i)$ is attached to the base platform with its x_i axis aligned with the rotary actuator in the x_i direction, and its z_i axis perpendicular to the fixed platform. x_i is rotated by γ_i from the X direction of fixed platform coordinate $A(O, X, Y, Z)$. The rotary actuators are located at the positions A_i (for $i = 1, 2, 3$) on the base platform and each shaft is connected to the lower part of the linear actuators through a universal joint (Figure 1). The upper parts of linear actuators are connected to the moving platform, B_i points, through spherical joints (Figure 1).

Cartesian coordinates $A(O, x, y, z)$ and $B(P, u, v, w)$ represented by $\{A\}$ and $\{B\}$ are attached to the base and moving platforms, respectively. In Figure 1, s_i represents the unit vector along the axes of i -th rotary actuator and d_i is the vector along A_iB_i with the length of d_i . Assuming that each limb is connected to the fixed base by a universal joint, the orientation of i -th limb with respect to the fixed base can be described by two successive rotations, rotation θ_i around the axis s_i , followed by rotation ψ_i around n_i , which is itself perpendicular to both d_i and s_i . It is to be noted that θ_i and d_i are active joints, actuated by the rotary and linear actuators, respectively, while, ψ_i is an inactive joint.

2.2. Experimental Evaluation

The performance of the Wide-Open mechanism was evaluated in an experimental study on a fully functional prototype of the robot. As shown in Figures 2 and 3, this prototype has full-circular fixed and moving platforms. Each of the three legs is equipped with a rotary actuator located on the base platform. This rotary actuator consist of a stepper motor followed by a low-ratio gearbox to enlarge the shaft torque. The gearbox shaft is then connected to the lower part of the leg using a revolute joint which is made of a miniature needle bearing. Linear actuation at each leg is provided using a ball screw system powered by another smaller stepper motor. Finally, the spherical joints connect the upper parts of the legs to the moving platform.

In the experimental tests, an optical stereoscopic vision system (Micron-Trackers, Claron Technology Inc., Ontario, Canada) was used as an external ob-

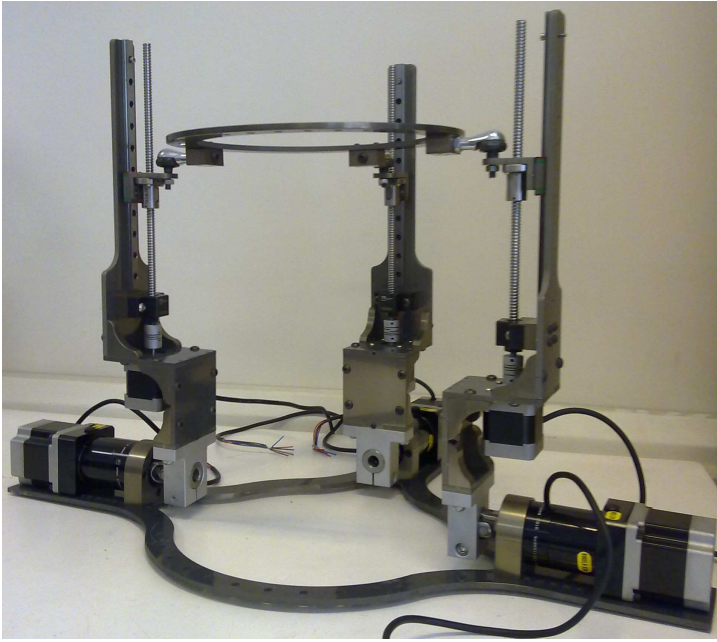


Figure 2. The fabricated prototype of the 3-legged Wide-Open robot. Full rings have been used for both fixed and moving platforms. At each leg, the ball screw actuator, driven by a stepper motor, is providing the precise linear motion. The rotary motion of each leg is generated by another stepper motor, attached to a low-aspect-ratio gearbox.

server to track the robot's end-effector, i.e., the center of the moving platform. In the setup, the 3D camera detects a set of markers attached to the moving platform to find its position and orientation with respect to a fixed Cartesian coordinate system, defined using another set of markers attached to the frame.

In the experimental tests, at first, the workspace of the robot was measured by running its actuators in different directions to the end of the moving range, and recording the end-effector position by the optical tracker. Trajectory tracking experiments were performed to verify the robot's kinematical model and to assess the effects of the friction and backlash. In each test, a desired trajectory, e.g., an offset circle, was defined for the end-effector to be followed and the corresponding joint trajectory for each actuator was computed, using the inverse kinematic equations, to generate the position data of the actuators. The

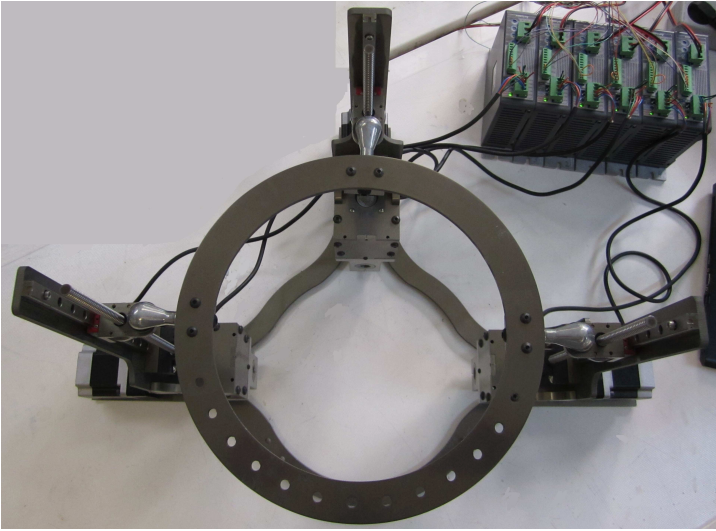


Figure 3. Top view of the fabricated Wide-Open robot, Figure 2. Since the legs are positioned in one half plane, the other half plane remains open. This non-symmetric placement of the legs, increases the workspace and gives a wide usable space between the two plates.

trajectory tracking experiments of the robot revealed reasonably acceptable results. The robot was capable of moving the end-effector on a 40 (mm) radius circle with a maximum deviation from the reference trajectory of about 8%.

3. Kinematic Analysis

One of the most important issues in the study of parallel manipulators is the kinematic analysis, where the generated results determine the applicability of the designed mechanism. In what follows, the inverse kinematic analysis of the parallel manipulator is formulated. It is to be noted that, the forward kinematic problem of this robot has 16 solutions at most, and its complete analysis is out of scope of this chapter.

Referring to Figure 1, $\mathbf{a}_i$ which represents OA_i can be written as

$$\mathbf{a}_i = g [\cos \gamma_i \quad \sin \gamma_i \quad 0]^T,$$

where g and h are the radius of the fixed and moving platforms, respectively. ${}^B\mathbf{b}_i$ represents the position of the i -th joint on the platform in the moving frame $\{B\}$, ${}^B\mathbf{b}_i = PB_i)_B$. ${}^B\mathbf{b}_i$ is constant and is equal to ${}^B\mathbf{b}_i = h[\cos\gamma_i \sin\gamma_i \ 0]^T$. We can represent ${}^A_B R = [r_{ij}]$, the rotation matrix from A to B , using Euler angles as

$${}^A_B R = \begin{bmatrix} c\alpha_2 c\alpha_3 & -c\alpha_2 s\alpha_3 & s\alpha_2 \\ c\alpha_3 s\alpha_2 s\alpha_1 + s\alpha_3 c\alpha_1 & -s\alpha_3 s\alpha_2 s\alpha_1 + c\alpha_3 c\alpha_1 & -c\alpha_2 s\alpha_1 \\ -c\alpha_3 s\alpha_2 c\alpha_1 + s\alpha_3 s\alpha_1 & s\alpha_3 s\alpha_2 c\alpha_1 + c\alpha_3 s\alpha_1 & c\alpha_2 c\alpha_1 \end{bmatrix}, \quad (1)$$

where $s\alpha_1 = \sin\alpha_1$ and $c\alpha_1 = \cos\alpha_1$, and so on. α_1 , α_2 , and α_3 are three Euler angles defined according to the $x-y-z$ convention. Thus, the vector ${}^B\mathbf{b}_i$ would be expressed in the fixed frame $\{A\}$ as

$$\mathbf{b}_i = {}^A_B R PB_i)_B. \quad (2)$$

Let $\mathbf{p}$ and $\mathbf{r}_i$ denote the position vectors for P and B_i in the reference frame $\{A\}$, respectively. From the geometry, it is obvious that

$$\mathbf{r}_i = \mathbf{p} + \mathbf{b}_i. \quad (3)$$

Subtracting vector $\mathbf{a}_i$ from both sides of (3) one obtains

$$\mathbf{r}_i - \mathbf{a}_i = \mathbf{p} + \mathbf{b}_i - \mathbf{a}_i. \quad (4)$$

Left hand side of (4) is the definition of $\mathbf{d}_i$, therefore

$$\mathbf{d}_i^2 = (\mathbf{p} + \mathbf{b}_i - \mathbf{a}_i) \cdot (\mathbf{p} + \mathbf{b}_i - \mathbf{a}_i). \quad (5)$$

Using Euclidean norm d_i can be expressed as

$$d_i = \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2}, \quad (6)$$

in which

$$\begin{cases} x_i = -h(\cos\gamma_i r_{11} + \sin\gamma_i r_{21}) + g \cos\gamma_i \\ y_i = -h(\cos\gamma_i r_{12} + \sin\gamma_i r_{22}) + g \sin\gamma_i \\ z_i = -h(\cos\gamma_i r_{13} + \sin\gamma_i r_{23}) \end{cases}. \quad (7)$$

Coordinates $C_i(A_i, x_i, y_i, z_i)$ are connected to the base platform with their x_i axes aligned with the rotary actuators in the $\mathbf{s}_i$ directions, with their z_i axes perpendicular to the fixed platform. Thus, one can express vector $\mathbf{d}_i$ in $\{C_i\}$ as

$${}^{C_i}\mathbf{d}_i = d_i \begin{bmatrix} \sin\psi_i \\ -\sin\theta_i \cos\psi_i \\ \cos\theta_i \cos\psi_i \end{bmatrix}, \quad (8)$$

and from the geometry is clear that

$$\mathbf{r}_i = \mathbf{a}_i + {}^A_{C_i}R^{C_i} \mathbf{d}_i, \quad (9)$$

where ${}^A_{C_i}R$ is the rotation matrix from $\{C_i\}$ to $\{A\}$,

$${}^A_{C_i}R = \begin{bmatrix} \cos \gamma_i & -\sin \gamma_i & 0 \\ \sin \gamma_i & \cos \gamma_i & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (10)$$

By equating the right sides of (3) and (4), and solving the resultant equation, ψ_i and θ_i can be calculated as follows:

$$\psi_i = \sin^{-1} \left(\frac{\cos \gamma_i (x - x_i) + \sin \gamma_i (y - y_i)}{d_i} \right), \quad (11)$$

and

$$\theta_i = \sin^{-1} \left(\frac{\sin \gamma_i (x - x_i) - \cos \gamma_i (y - y_i)}{d_i \cos \psi_i} \right). \quad (12)$$

4. Workspace Analysis

Consider the Wide-Open mechanism with $g = 1$ (m) and $h = 0.5$ (m), where g and h are the radii of the fixed and moving platforms, respectively. By assuming a cubic with 1 (m) length, 1 (m) width and 1 (m) height located 0.25 (m) above the base platform, we are interested in determining the volume percentage in which the mechanism can successfully reach the locations within this cubic space.

The experimental workspace of the Wide-Open robot is illustrated in Figure 4. There were two main mechanical constraints in the fabricated prototype that limited its workspace. The first constraint was the length change of the linear actuators, d_i , being between 0.22 (m) to 0.40 (m). The other restriction was imposed by the spherical joints, θ_i^{SJ} . The spherical joints, used in our prototype, could rotate only up to ± 25 (deg). The workspace of the mechanism can be enlarged to the theoretical one if these constraints are removed. Figure 4 also illustrates the two potential enlarged workspaces of the mechanism; one with the increased spherical rotation limits of ± 50 (deg), and the other with an ideal spherical joint without a rotational constraint.

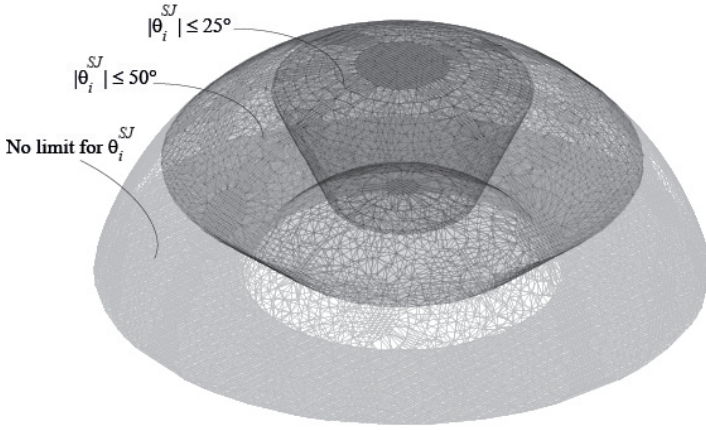


Figure 4. Workspaces of the Wide-Open parallel mechanism with different constraints on the spherical joint angles. Spherical joints are considered to have no limit, restricted to rotate up to ± 50 (deg), and to ± 25 (deg); respectively.

5. Dynamic Analysis

5.1. Inverse Dynamics

Due to the numerous constraints imposed by closed loops of a parallel manipulator, deriving explicit equations of motion in terms of a set of independent generalized coordinates become a prohibitive task. In this regard, the principle of virtual work appears to be the most efficient method of analysis. In this section we apply the principle of virtual work to derive a transformation between joint torques/forces and end-effector forces/moments.

For convenience, we introduce a six-dimensional wrench $\hat{\mathbf{F}}_i$ as the sum of applied and inertia wrenches about the center of mass of link i . Similarly, we assume $\hat{\mathbf{F}}_p$ to denote the sum of applied and inertia wrenches about the center of mass of the moving platform. Then the principle of virtual work for a parallel manipulator can be stated as

$$\delta \mathbf{q}^T \boldsymbol{\tau} + \delta \mathbf{x}^T \hat{\mathbf{F}}_p + \sum_i \delta \mathbf{x}_i^T \hat{\mathbf{F}}_i = 0, \quad (13)$$

where, $\boldsymbol{\tau}$ is the wrench applied to the actuators. $\delta \mathbf{x}$ is the virtual displacement and rotation of the center of the moving platform. $\delta \mathbf{x}_i$ is the virtual displacement and rotation of the center of link i , and $\delta \mathbf{q}$ is the virtual displacement and

rotation of $\mathbf{q}$. One can relate $\delta \mathbf{q}$ and $\delta \mathbf{x}$ as follows

$$\delta \mathbf{q} = J_p \delta \mathbf{x}, \quad J_p = J_q^{-1} J_x. \quad (14)$$

By introducing J_i as link's jacobian matrix, we have

$$\delta \mathbf{x}_i = J_i \delta \mathbf{x}. \quad (15)$$

Substituting (14) and (15) into (13), results in

$$J_p^T \boldsymbol{\tau} + \hat{\mathbf{F}}_p + \sum_i J_i^T \hat{\mathbf{F}}_i = 0. \quad (16)$$

If J_p is a square matrix, solving for $\boldsymbol{\tau}$ in (16) is straight forward, which results in

$$\boldsymbol{\tau} = -J_p^{-T} \left(\hat{\mathbf{F}}_p + \sum_i J_i^T \hat{\mathbf{F}}_i \right). \quad (17)$$

It is to be noted that, errors in the system parameters may result in deviation of the platform position and orientation from those desired. Therefore, in Section 6, a robust sliding-mode controller is designed to cancel out the effects of parameters uncertainties and external load disturbances.

5.2. Forward Dynamics

In this section, the forward dynamic analysis of the Wide-Open parallel robot is summarized. The robot is a 6-DOF closed kinematic chain mechanism consisting of a fixed base, a moving platform, and 3 legs connecting the two platforms. The basic procedure in deriving the forward dynamic model is using Newton-Euler method, which incorporates the constraint forces in a compact form. The general equations of motion for the mechanism, in the workspace coordinates, could be written as [2]

$$M(\mathbf{x})\ddot{\mathbf{x}} + H(\mathbf{x}, \dot{\mathbf{x}}) = J_p^T \boldsymbol{\tau}, \quad (18)$$

where $M(\mathbf{x})$ is the 6×6 mass matrix and $H(\mathbf{x}, \dot{\mathbf{x}})$ is a 6×1 nonlinear vector including coriolis, centrifugal and gravitational effects. Forward dynamic problem would be to solve (18) to obtain $\ddot{\mathbf{x}}$ as

$$\ddot{\mathbf{x}} = -M(\mathbf{x})^{-1}H(\mathbf{x}, \dot{\mathbf{x}}) + M(\mathbf{x})^{-1}J_p^T \boldsymbol{\tau}. \quad (19)$$

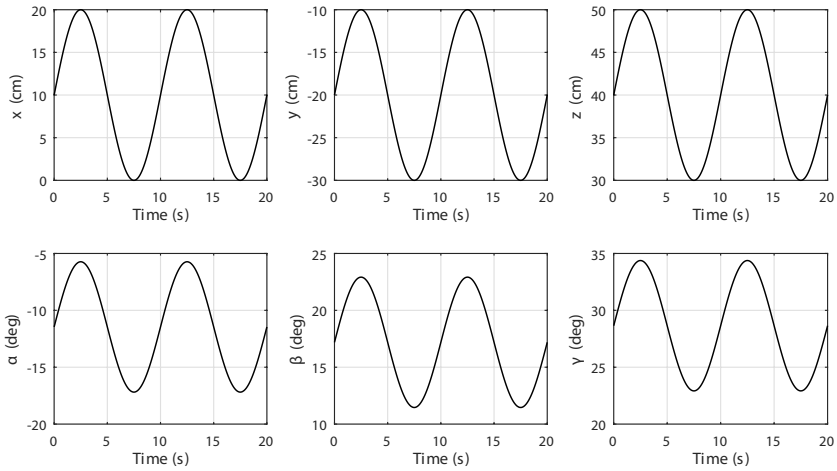


Figure 5. Reference trajectory of the center of the moving platform. Frequency of the sinusoidal movements is $2\pi/10$. x , y , and z are translational motions and α , β , and γ are the roll, pitch, and yaw angles; respectively.

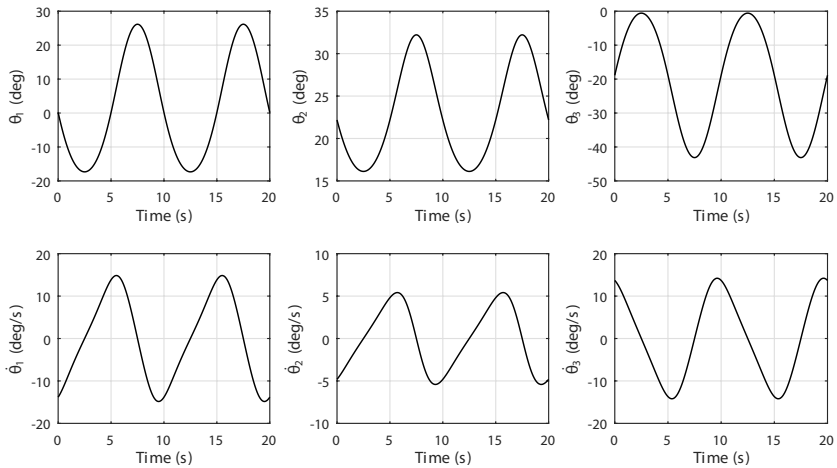


Figure 6. Time history of the angular rotations, θ_i , and angular velocities, $\dot{\theta}_i$, of the rotary actuators, based on the reference trajectory in Figure 5.

5.3. Dynamic Simulations

In this section, we numerically investigate the inverse dynamic solution of the Wide-Open mechanism. The mass of the moving platform is 0.3 (kg). The

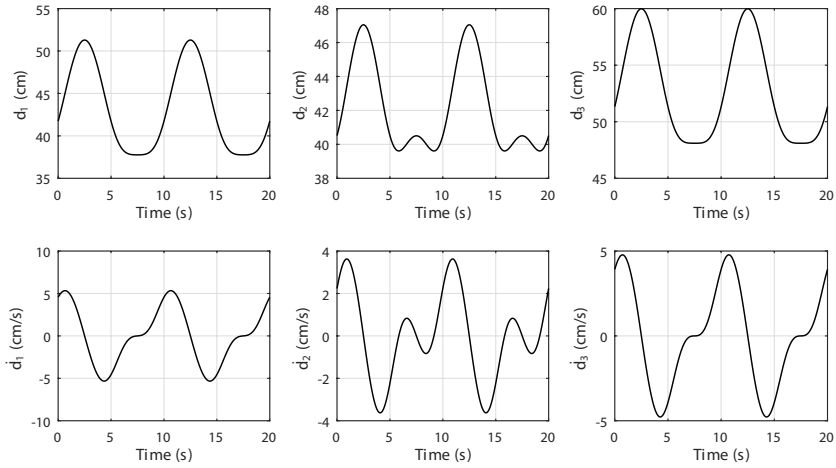


Figure 7. Time history of the linear displacements, d_i , and linear velocities, $\dot{d}_i$, of the linear actuators, based on the reference trajectory in Figure 5.

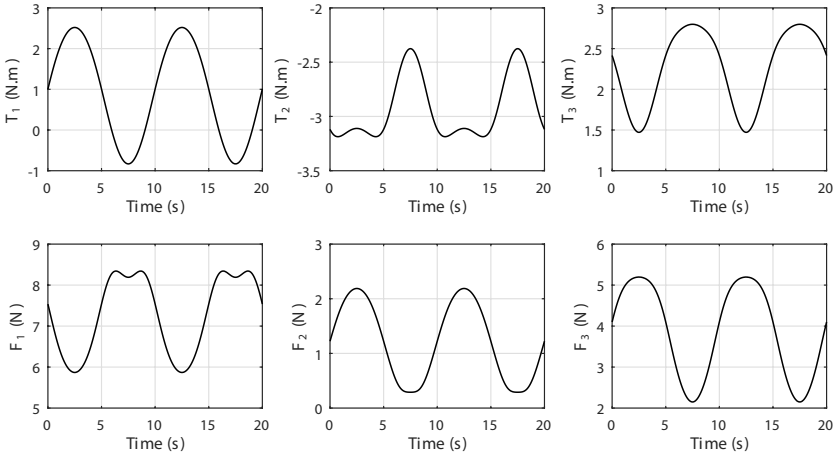


Figure 8. Inverse dynamic solution, i.e., actuators' torques and forces, based on the reference trajectory in Figure 5.

masses of the lower part and upper part of each leg are 0.9 (kg) and 0.4 (kg); respectively. radii of the moving platform and the fixed platform are 0.14 (m) and 0.18 (m); respectively. The input reference trajectory is shown in Figure

5. As shown in the figure, the oscillations have the same frequency of $2\pi/10$, in all translational and rotational degrees of freedom. All three translational movements have the same amplitudes of 0.1 (m), while their initial conditions are different. Similarly, the rotational movements have the same amplitudes of 0.1 (rad) or 5.73 (deg), with different initial conditions.

Based on the same reference trajectory, time history of the rotations of the three active rotary actuators are shown in Figure 6. Both angular rotations, θ_i , and angular velocities, $\dot{\theta}_i$, are shown in the figure. Time history of the displacements of the three linear actuators are shown in Figure 7. Linear displacements, d_i , and linear velocities, $\dot{d}_i$, are shown in the figure.

Inverse dynamic solution of the reference trajectory of Figure 5, is shown in Figure 8. The actuators' torques and forces, τ , are calculated from Eq. (17). It is to be noted that, the results for the two time periods of 10 (s) are identical.

6. Position Control Analysis

6.1. Sliding-Mode Control Design

In this section, the sliding-mode control technique, which is an effective way for controlling nonlinear systems with un-modeled uncertainties [4], is used to develop a controller allowing tracking of the reference inputs for the 6-DOF Wide-Open mechanism. Desired Cartesian trajectories are used as input commands to the controller. The controller then sets the actuators torques and forces in the real-time.

In absence of disturbances and uncertain terms, the control problem would be easily accomplished by replacing τ from inverse dynamic solution, (17), into (19). However, in real situations, one should consider the effects of disturbances and uncertainties by introducing the 6×1 vector Δ into (19) as

$$\ddot{\mathbf{x}} = -M(\mathbf{x})^{-1}H(\mathbf{x}, \dot{\mathbf{x}}) + M(\mathbf{x})^{-1}J_p^T \tau + \Delta, \quad (20)$$

in which $\Delta = [\Delta_1, \Delta_2, \Delta_3, \Delta_4, \Delta_5, \Delta_6]^T$, with the following boundaries,

$$|\Delta_i| \leq \delta_i, \quad (21)$$

for $i = 1$ to 6. It is to be noted that, it would be difficult to simultaneously guarantee the accuracy and robustness of a controller, when the system load is uncertain and varying. A lot of research studies design robust controllers to deal

with the external disturbances. However, it depends on the assumptions that the external disturbances are bounded [4].

To derive input actuators forces and torques, τ , for the general system (20), one can use the following definition,

$$\tau = \tau^{inv} - \tau^\Delta, \quad (22)$$

where τ^{inv} stands for the input forces and torques required for the nominal system, (19), and τ^Δ is present to cancel the effects of the disturbances and uncertainties, Δ . Replacing (22) into (20), one obtains

$$\ddot{\mathbf{x}} = \ddot{\mathbf{x}}^{des} + \Delta - M(\mathbf{x})^{-1} J_p^T \tau^\Delta, \quad (23)$$

in which, $\mathbf{x}^{des}$ is the desired 6×1 vector of position and orientation of the center of the moving platform. By defining W as

$$W = M(\mathbf{x})^{-1} J_p^T \tau^\Delta, \quad (24)$$

τ^Δ will be obtained as

$$\tau^\Delta = J_p^{-T} M(\mathbf{x}) W. \quad (25)$$

Replacing (24) into (23) yields to

$$\ddot{\mathbf{x}} = \ddot{\mathbf{x}}^{des} + \Delta - W. \quad (26)$$

The displacement error variable $\mathbf{z}_i$, and its derivative $\mathbf{z}_i^d$, are respectively defined as,

$$\mathbf{z}_i = \mathbf{x}_i - \mathbf{x}_i^{des}, \quad (27)$$

and

$$\mathbf{z}_i^d = \dot{\mathbf{x}}_i - \dot{\mathbf{x}}_i^{des}. \quad (28)$$

Using Eqs. (26), (27), and (28), we can reach to the following second-order state-space system,

$$\begin{cases} \dot{\mathbf{z}}_i = \mathbf{z}_i^d, \\ \dot{\mathbf{z}}_i^d = \Delta_i - W_i. \end{cases} \quad (29)$$

One can define the sliding surface S_i as follows,

$$S_i = (d/dt + \lambda_i) z_i = \dot{z}_i^d + \lambda_i z_i, \quad (30)$$

where the λ_i parameter is chosen based on the desired convergence speed of the controller. The derivative of the sliding surface is obtained as,

$$\dot{S}_i = \Delta_i - W_i + \lambda_i z_i^d. \quad (31)$$

In what follows, we use saturation function, $\text{sat}(\cdot)$, for omitting the noises generated from the $\text{sign}(\cdot)$ function,

$$W_i = W_i^{eq} + \beta_i \text{sat}(S_i/\phi_i), \quad (32)$$

in which, smaller values for ϕ_i result in lower errors by increasing the oscillations, while larger values for ϕ_i increase the error and decrease the oscillations. One may use the following formulation for ϕ_i selection based on the maximum resultant error of displacement, z_i^{max} ,

$$\lambda_i z_i^{max} \leq \phi_i. \quad (33)$$

According to the definition, W_i^{eq} is calculated from (31) by setting $\dot{S}_i$ and Δ_i to zero,

$$W_i^{eq} = \lambda_i z_i^d. \quad (34)$$

We define the Lyapunov function as,

$$V_i = \frac{S_i^2}{2}, \quad (35)$$

where its derivative is bounded by

$$\dot{V}_i = S_i \dot{S}_i \leq -\eta_i |S_i|. \quad (36)$$

By replacing Eqs. (30) and (31) into (36), and after simplifications, we reach to the following condition for β_i ,

$$\delta_i + \eta_i \leq \beta_i. \quad (37)$$

As a result, all the design parameters have defined boundaries. Finally, by replacing (32) into (25), the vector of input forces and torques due to existence of disturbances and uncertainties is determined. The general algorithm of the proposed closed-loop sliding-mode controller is illustrated in Figure 9.

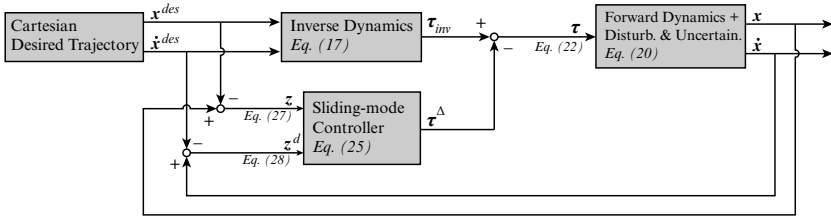


Figure 9. Proposed closed-loop tracking control schematics. The sliding-mode controller generates the required forces and torques to cancel out the negative effects of the disturbances and uncertainties on the system.

6.2. Position Control Simulations

To verify the effectiveness and robustness of the proposed control algorithm, the tracking performance of the controller is evaluated in this section. The proposed control method is numerically evaluated in three case studies. The control parameters λ_i , ϕ_i , and β_i are respectively set to 5, 2, and 50 through all the numerical simulations.

6.2.1. Initial Condition Alteration

Initial condition alteration, is a classic way of analyzing the robustness and performance of a controller. In this section, we have altered the initial position of each of x , y , and z translational degrees of freedom by 0.2 (m) from that of the reference trajectory in Figure 5. As seen from the Figure 10, the controller is well capable of following the reference trajectory with zero error, after the first few seconds. The corresponding actuators' forces and torques are shown and compared to those of the reference trajectory in Figure 11. As expected, an overshoot is seen right after the start point, then the forces and torques rapidly converge to those of the reference trajectory.

6.2.2. Parameters Uncertainties

As another simulation, the robustness of the controller can be demonstrated by applying the controller to the system with different parameter values. Parameters uncertainties could be well modeled using this approach. To this end, every component's mass in the system is considered to be 20% larger than that reported and initiated in the controller. This way, the controller will be faced to

the presence of a relatively large parametric uncertainty. The trajectory tracking results are shown in Figure 12. As seen in the figure, even though there is a strong parametric uncertainty in the system, the controller has successfully tracked the reference trajectory in a satisfactory fashion. It is to be noted that, unlike the initial condition alteration, the parametric difference of 20% is always present in the controller, and therefore the output trajectory can not converge to the reference trajectory after awhile. The corresponding actuators' forces and torques are shown and compared to those of the reference trajectory in Figure 13. As seen in the figure, an offset is seen between most pairs of the gray and black lines. This simulation, illustrates the robustness of the proposed sliding-mode controller in the existence of parameters uncertainties.

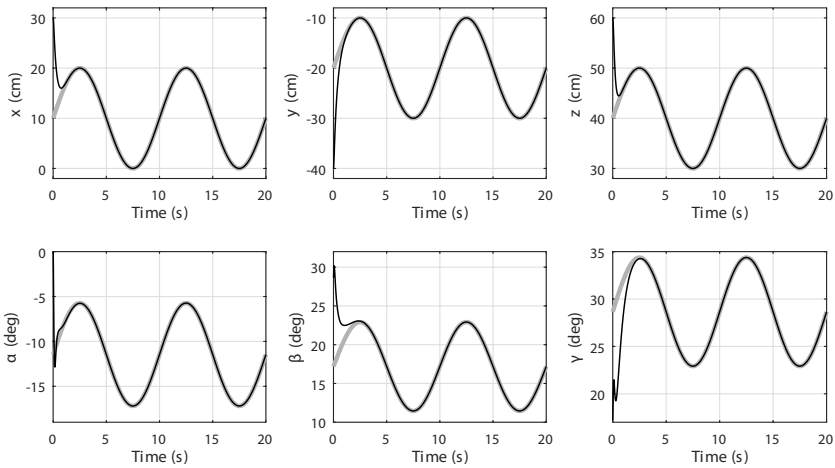


Figure 10. Initial condition of the system is altered with that of the reference trajectory. The initial position of each of x , y , and z differs 0.2 (m) with that of the reference trajectory. The initial orientation of each of α , β , and γ differs 0.2 (rad) or 11.46 (deg), with that of the reference trajectory. Black line: output trajectory of the system. Gray line: reference trajectory, i.e., Figure 5.

6.2.3. External Disturbances

Load disturbances are the main external disturbances of the parallel manipulators, and greatly impact the system performance [4]. To study the robustness of the proposed controller when an external disturbance occurs, the moving plat-

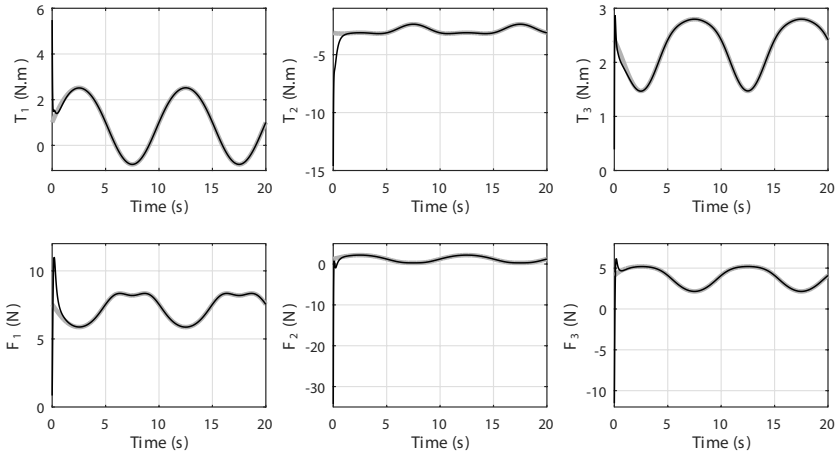


Figure 11. Black line: actuators' forces and torques corresponding to the initial condition change, i.e., Figure 10. Gray line: actuators' forces and torques corresponding to the reference trajectory, i.e., Figure 8.

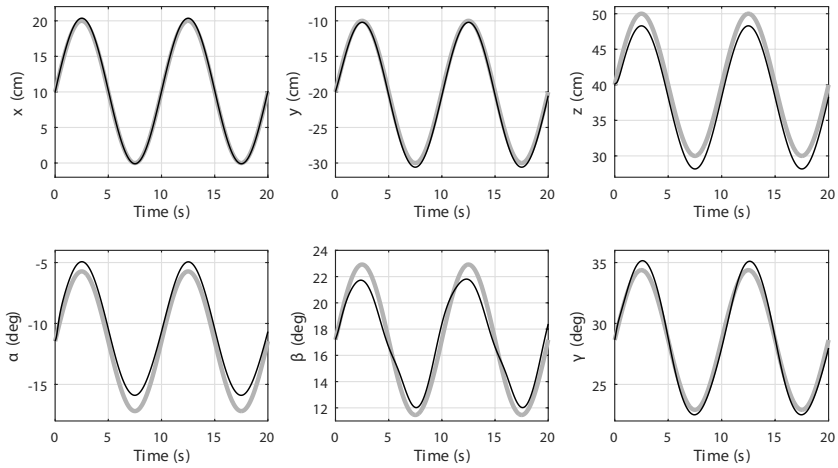


Figure 12. Mass of the components of the system is increased by 20% compared to the data initiated in the controller model. Black line: output trajectory of the system. Gray line: reference trajectory, i.e., Figure 5.

form is hit with an external load from three directions of x , y , and z . The load

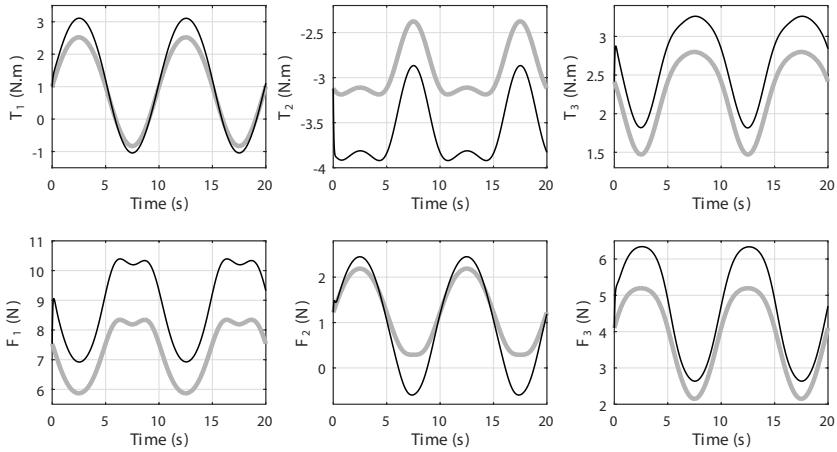


Figure 13. Black line: actuators' forces and torques corresponding to the mass change simulation, i.e., Figure 12. Gray line: actuators' forces and torques corresponding to the reference trajectory, i.e., Figure 8.

hits the platform at $t = 5$ (s) and is detached after 0.2 (s). The magnitude of the load is 9.81 (N), in each direction. The results of the simulation is shown in Figure 14. The corresponding actuators' forces and torques are shown and compared to those of the reference trajectory in Figure 15. It is obvious from the figures that the controller is able to cancel out the sudden impact of the external disturbance by properly overshooting the actuators' forces and torques, right after the incident. Shortly after, the output trajectory is identical with the reference trajectory.

Conclusion

The 6-DOF 3-legged Wide-Open parallel mechanism was analyzed. In comparison with the well-known Gough-Stewart platform, the passive universal joints are replaced with active joints, reducing the number of legs from 6 to 3. Since the active rotary actuators are resting on the fixed plate, the moving platform of the robot can achieve higher accelerations, due to the smaller inertial effects.

The development of a sliding-mode tracking controller for the 6-DOF Wide-Open mechanism was investigated, using the dynamic model of the parallel robot. Lyapunov analysis approach was used in designing the sliding-mode

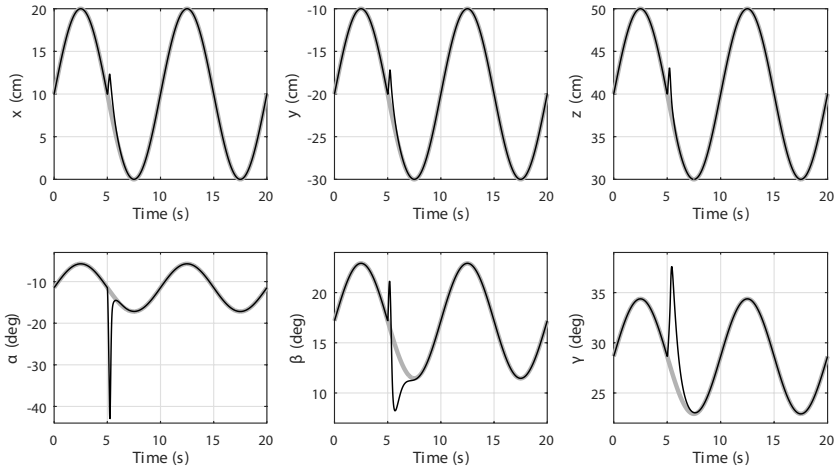


Figure 14. An external load hits the moving platform in all three x , y , and z directions. The load hits the platform at $t = 5$ (s) and is detached after 0.2 (s). The magnitude of the load is 9.81 (N), in each direction. Black line: output trajectory of the system. Gray line: reference trajectory, i.e., Figure 5.

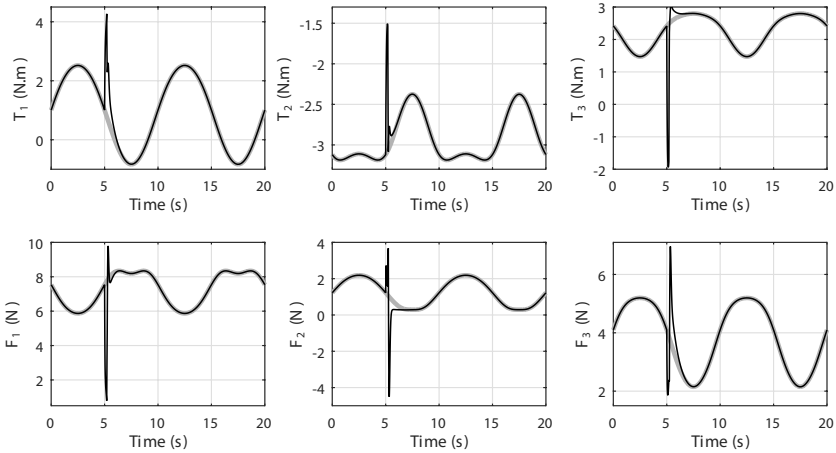


Figure 15. Black line: actuators' forces and torques corresponding to the external load impact, i.e., Figure 14. Gray line: actuators' forces and torques corresponding to the reference trajectory, i.e., Figure 8.

controller. The trajectory tracking control of the 6-DOF parallel robot with uncertain load disturbances and parameter uncertainties was demonstrated through three numerical simulations. The simulation results show that the proposed nonlinear control strategy is robust in presence of disturbances and uncertainties.

References

- [1] B. Dasgupta, T. Mruthunjaya, The stewart platform manipulator: a review, *Mechanism and machine theory* 35 (1) (2000) 15–40.
- [2] H. Guo, Y. Liu, G. Liu, H. Li, Cascade control of a hydraulically driven 6-dof parallel robot manipulator based on a sliding mode, *Control Engineering Practice* 16 (9) (2008) 1055–1068.
- [3] I. Davliakos, E. Papadopoulos, Model-based control of a 6-dof electrohydraulic stewart–gough platform, *Mechanism and machine theory* 43 (11) (2008) 1385–1400.
- [4] Y. Pi, X. Wang, Trajectory tracking control of a 6-dof hydraulic parallel robot manipulator with uncertain load disturbances, *Control Engineering Practice* 19 (2) (2011) 185–193.
- [5] V. E. Gough, S. G. Whitehall, Universal tyre test machine, in Proc 9th Int. Tech. Congress FISITA (1962) 117–137.
- [6] D. Stewart, A platform with six degrees of freedom, *Proc. Inst. Mech. Eng.* 180 (1965) 371–386.
- [7] B. M. St-Onge, C. M. Gosselin, Singularity analysis and representation of the general gough-stewart platform, *Int. J. Rob. Res.* 19 (3) (2000) 271–288.
- [8] H. Li, C. M. Gosselin, M. J. Richard, Determination of the maximal singularity-free zones in the six-dimensional workspace of the general, *Mech. Mach. Theory* 42 (4) (2007) 497–511.
- [9] Q. Jiang, C. M. Gosselin, Determination of the maximal singularity-free orientation workspace for the gough-stewart platform, *Mech. Mach. Theory* 44 (6) (2009) 1281–1293.

-
- [10] Q. Jiang, C. M. Gosselin, Maximal Singularity-Free Total Orientation Workspace of the GoughStewart Platform, *J. Mech. Rob.*, Trans. ASME 1 (3) (2009) 1–4.
- [11] Q. Jiang, C. M. Gosselin, Evaluation and Representation of the Theoretical Orientation Workspace of the GoughStewart Platform, *J. Mech. Rob.*, Trans. ASME 1 (2) (2009) 1–9.
- [12] B. Inner, S. Kucuk, A novel kinematic design, analysis and simulation tool for general stewart platforms, *Simulation* (2013) 0037549713482733.
- [13] G. Liu, Z. Qu, X. Liu, J. Han, Singularity analysis and detection of 6-ucu parallel manipulator, *Robotics and Computer-Integrated Manufacturing* 30 (2) (2014) 172–179.
- [14] A. Karimi, M. T. Masouleh, P. Cardou, Singularity-free workspace analysis of general 6-ups parallel mechanisms via convex optimization, *Mechanism and Machine Theory* 80 (2014) 17–34.
- [15] W. Zhou, W. Chen, H. Liu, X. Li, A new forward kinematic algorithm for a general stewart platform, *Mechanism and Machine Theory* 87 (2015) 177–190.
- [16] B. Monsarrat, C. M. Gosselin, Workspace analysis and optimal design of a 3-leg 6-dof parallel platform mechanism, *Robotics and Automation*, IEEE Transactions on 19 (6) (2003) 954–966.
- [17] M. Badescu, C. Mavroidis, Workspace optimization of 3-legged upu and ups parallel platforms with joint constraints, *Journal of Mechanical Design* 126 (2) (2004) 291–300.
- [18] Y. Lu, Y. Shi, Z. Huang, J. Yu, S. Li, X. Tian, Kinematics/statics of a 4-dof over-constrained parallel manipulator with 3 legs, *Mechanism and Machine Theory* 44 (8) (2009) 1497–1506.
- [19] C. Zhang, L. Zhang, Kinematics analysis and workspace investigation of a novel 2-dof parallel manipulator applied in vehicle driving simulator, *Robotics and Computer-Integrated Manufacturing* 29 (4) (2013) 113–120.
- [20] G. Coppola, D. Zhang, K. Liu, A new class of adaptive parallel robots, *Journal of Mechanisms and Robotics* 6 (4) (2014) 041013.

- [21] H. Azulay, M. Mahmoodi, R. Zhao, J. K. Mills, B. Benhabib, Comparative analysis of a new $3 \times$ pprs parallel kinematic mechanism, *Robotics and Computer-Integrated Manufacturing* 30 (4) (2014) 369–378.
- [22] A. Ghobakhloo, M. Eghtesad, M. Azadi, Position control of a stewart-gough platform using inverse dynamics method with full dynamics, in: *Advanced Motion Control, 2006. 9th IEEE International Workshop on, IEEE, 2006*, pp. 50–55.
- [23] M. R. Sirouspour, S. E. Salcudean, Nonlinear control of hydraulic robots, *Robotics and Automation, IEEE Transactions on* 17 (2) (2001) 173–182.
- [24] J. Rosario, E. Oliveira, D. Dumu, Conception of stewart-gough platform with reconfigurable control using integrate prototyping, in: *Power Electronics, Electrical Drives, Automation and Motion, 2006. SPEEDAM 2006. International Symposium on, IEEE, 2006*, pp. 1039–1044.
- [25] M. H. Abedinnasab, Y.-J. Yoon, H. Zohoor, Exploiting Higher Kinematic Performance—Using a 4-Legged Redundant *PM Rather than Gough-Stewart Platforms*, InTech, 2012.
- [26] M. H. Abedinnasab, G. R. Vossoughi, Analysis of a 6-dof redundantly actuated 4-legged parallel mechanism, *Nonlinear Dyn.* 58 (4) (2009) 611–622.
- [27] O. Aghababai, Design, Kinematic and Dynamic Analysis and Optimization of a 6 DOF Redundantly Actuated Parallel Mechanism for Use in Haptic Systems, MSc Thesis, Sharif University of Technology, Tehran, Iran (2005).

INDEX

#

2R2T, viii, ix, 81, 82, 83, 84, 85, 87, 88, 89,
91, 92, 93, 94, 95, 96, 97
6-DOF 3-legged Wide-Open parallel
manipulator, 146

A

actuation, 8, 37, 38, 87, 88, 97, 147
actuators, 9, 15, 19, 27, 29, 47, 50, 78, 82,
83, 97, 108, 110, 115, 116, 117, 118, 119,
120, 122, 123, 130, 132, 138, 145, 146,
147, 148, 150, 151, 152, 154, 155, 156,
157, 159, 160, 161, 162, 163
adaptation, vii, 1, 2, 3, 4, 5, 6, 16, 17, 18,
19, 20, 23, 28
adaptive control, vii, viii, 1, 2, 3, 4, 6, 7, 8,
16, 18, 21, 22, 24, 27, 32, 34, 35, 36, 37,
38
agility, 103, 128, 138
algorithm, 78, 139, 142, 145, 158, 165
amplitude, 31
Artificial Neural Networks, 5
automation, 102
autonomous navigation, 105

B

backlash, 63, 116, 148
bandwidth, 3, 19, 20
base, viii, 9, 10, 45, 55, 81, 82, 84, 88, 90,
104, 112, 144, 145, 147, 150, 151, 153

batteries, 105, 126, 127, 130
biped walking, 101, 103

C

CAD, 9, 10
calibration, 42, 63
case studies, 5, 158
case study, 141
CDC, 35
CENTAUROB, vii, ix, 101, 102, 103, 105,
106, 107, 108, 109, 110, 111, 112, 113,
114, 115, 116, 117, 119, 120, 121, 122,
123, 124, 125, 127, 128, 129, 131, 132,
133, 134, 135, 137, 138, 139, 140, 141
challenges, 42, 104, 105
closed-loop system, 1, 3, 4, 7, 19, 25, 71
closure, 48, 50
collisions, 117, 126, 129
comparative analysis, 65, 66
compensation, 6, 8, 35, 78, 80, 113, 116,
141
competitiveness, 109
complexity, 42, 70, 130
compliance, 112
computation, 78, 105, 139, 141
computed torque control, 41, 43, 63, 64, 65
computer, 27
configuration, 13, 68, 70, 72, 73, 74, 78, 95,
97, 110, 127, 141
connectivity, 88, 89, 96
constant load, 127
construction, ix, 101, 102, 105, 141
consumption, 127, 130, 132
control concepts, 101

convention, 45, 46, 150
 convergence, 2, 116, 157
 cost, 63, 64, 73, 75, 97, 116

D

damping, 5
 decoupling, 16, 35, 76
 Delta, 9, 14, 82, 140
 derivatives, 22, 46, 56, 57, 120
 detection, 105, 115, 164
 deviation, 70, 72, 73, 149, 153
 dichotomy, 104
 displacement, 87, 91, 110, 117, 118, 131,
 132, 152, 157, 158
 distribution, 67, 68, 70, 73, 104, 130
 dynamic control, 78
 dynamics, viii, ix, 1, 2, 3, 5, 6, 8, 14, 15, 16,
 17, 18, 19, 21, 22, 23, 25, 29, 32, 33, 45,
 47, 59, 66, 70, 71, 75, 76, 77, 79, 82, 102,
 103, 112, 141, 144, 165

E

electricity, 115, 127
 energy, 31, 55, 57, 58, 105, 127
 energy consumption, 105
 engineering, 112
 environment, ix, 2, 4, 101, 111, 133, 139
 environments, 102, 104, 105, 106, 134, 135,
 136, 137
 equilibrium, 119
 equipment, 104, 144
 evolution, 4, 5, 8, 17, 23, 27, 31, 69, 140
 excitation, 112
 execution, 124, 125, 126
 extracts, 52

F

factories, vii, viii, 41
 filters, 27
 flexibility, 103, 104, 106, 108, 112, 116
 flight, 36, 82, 144
 force, 14, 15, 75, 87, 89, 97, 111, 115, 119,
 125, 130

freedom, viii, 43, 44, 45, 47, 77, 81, 82, 85,
 88, 98, 99, 104, 105, 108, 110, 111, 113,
 114, 115, 118, 119, 132, 136, 140, 141,
 156, 159, 164
 friction, 8, 14, 25, 44, 63, 66, 68, 108, 113,
 148
 fully-isotropic, 83, 85, 87, 89, 91, 93, 94,
 95, 97, 99

G

geometry, 49, 98, 140, 142, 150, 151
 google, 38
 Gough-Stewart platform, 41, 43, 44, 45, 47,
 48, 54, 60, 64, 72, 75, 104, 144, 161
 gravitation, 62
 gravitational effect, 153
 gravitational force, 16
 gravity, ix, 14, 15, 43, 55, 59, 70, 78, 80,
 101
 guidance, 104

H

height, 106, 109, 115, 124, 125, 128, 129,
 130, 131, 132, 136, 137, 139, 151
 hybrid, viii, 81, 82, 83, 84, 85, 86, 89, 96

I

ICRA, 36, 37, 38, 39, 140, 141
 industrial environments, 102
 industry, 42, 106, 134
 inertia, 3, 15, 43, 55, 57, 58, 59, 66, 67, 70,
 71, 72, 103, 152
 inertial effects, 146, 163
 intelligence, 102
 interference, 42, 117, 126
 iteration, 67

J

joints, viii, ix, 2, 9, 10, 11, 13, 14, 15, 16,
 24, 42, 44, 47, 48, 49, 51, 62, 64, 66, 68,
 70, 72, 75, 82, 83, 84, 85, 86, 87, 88, 89,
 90, 91, 92, 94, 95, 96, 97, 103, 104, 105,

106, 107, 108, 112, 113, 114, 116, 118,
122, 145, 146, 147, 151, 152, 161, 163

K

kinematic chain structure, vii, viii, 41, 42
kinematic walking robot, ix, 101, 102

L

L1 adaptive control, vii, 3, 16, 17, 19, 21,
23, 24, 28, 29, 30, 32, 33, 36, 38
Lagrange multipliers, 43, 59
Lagrangian formulation, 43, 54
laws, 2, 19, 23, 27, 42, 62, 75, 145
legs, ix, 59, 82, 83, 84, 101, 105, 106, 108,
113, 114, 115, 117, 127, 144, 145, 146,
147, 149, 153, 163, 165
linear systems, 37
lithium, 127
localization, 138
Lyapunov function, 158

M

magnet, 117
magnitude, 160, 162
manufacturing, 104, 117, 144
mapping, ix, 82, 86, 91
mass, vii, viii, 14, 15, 16, 41, 55, 66, 67,
105, 106, 111, 115, 119, 125, 130, 131,
132, 133, 152, 153, 154, 159, 162
matrix, ix, 3, 13, 14, 15, 16, 17, 20, 21, 24,
25, 26, 27, 28, 43, 45, 46, 52, 53, 54, 55,
57, 58, 64, 70, 71, 82, 83, 84, 86, 89, 91,
95, 97, 111, 112, 120, 121, 122, 123, 150,
151, 153
measurement, 8, 29, 43, 76, 78, 115, 116,
117
medical, 75, 82
methodology, 44, 49, 72, 78
modelling, viii, 41, 42, 43, 44, 45, 47, 49,
51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71,
73, 75, 77, 79, 141
models, 105, 106
modifications, 102, 129
modules, 113

morphology, 99
motion control, 35, 79, 103, 124, 126
motivation, 6, 33, 105
muscles, 106

N

neural networks, 5, 6, 37
nickel, 127
nonlinear dynamics, viii, 1, 2, 3, 6, 8, 33
nonlinear system, 1, 156

O

obstacles, ix, 101, 102, 112, 128, 132, 133,
136, 142
omission, 132
operations, 4, 104
optimization, 98, 112, 127, 144, 165

P

parallel, vii, viii, ix, 1, 2, 3, 4, 5, 6, 7, 8, 9,
10, 21, 22, 24, 32, 33, 36, 37, 38, 42, 43,
44, 46, 47, 59, 62, 63, 64, 70, 75, 76, 77,
78, 79, 80, 81, 82, 83, 84, 85, 87, 88, 89,
90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100,
101, 102, 103, 104, 106, 109, 112, 115,
128, 134, 137, 140, 141, 143, 144, 145,
146, 149, 152, 153, 160, 161, 163, 164,
165, 166
parallel manipulators, 1, 144
parallel platform, 144, 165
parallel robots, vii, viii, 41, 42, 101, 102,
103
parameter estimation, 76
patents, 38
path planning, viii, 81, 82, 97, 139, 142
pitch, 49, 52, 87, 88, 154
platform, viii, ix, 4, 5, 9, 13, 14, 15, 24, 36,
41, 42, 43, 44, 45, 47, 48, 49, 50, 54, 55,
58, 59, 60, 64, 66, 70, 72, 75, 76, 77, 79,
81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91,
94, 95, 97, 102, 104, 109, 111, 112, 116,
118, 119, 134, 137, 142, 144, 145, 146,
147, 148, 149, 150, 151, 152, 153, 154,
155, 157, 160, 161, 162, 163, 164, 165

polar, 55
 Portugal, 36
 position control, 41, 62, 125, 156, 158
 positioning errors, vii, viii, 42, 103, 144
 programming, 5
 project, 34, 71, 139
 proposition, 33
 prototype, vii, ix, 4, 33, 101, 102, 103, 106,
 112, 113, 114, 140, 141, 147, 148, 151

R

radar, 82
 radius, 10, 11, 12, 108, 149
 reciprocal screw theory, viii, 81, 83, 87, 96
 redundancy, viii, 8, 38, 43, 64, 97
 reference frame, 48, 49, 55, 150
 reference system, 17, 19
 rejection, 28, 79
 requirements, 2, 4, 42, 63, 67, 102, 108,
 112, 145
 resilience, 105, 112
 resolution, 42, 78, 141
 response, 22, 28
 retirement, 135, 136
 rings, 148
 robotic manipulators, vii, viii, ix, 35, 36, 81,
 82, 83, 87, 93, 96
 robotics, 42, 43, 44, 50, 75, 104, 140, 147
 rotational inertia, 14
 rotational matrix, 86
 rotations, 29, 114, 147, 154, 156

S

safety, 112, 134
 saturation, 158
 schema, 146, 159
 semicircle, 146
 sensitivity, viii, 43, 70, 72, 73, 74, 75, 79
 sensor architecture, 101
 sensors, viii, 29, 42, 43, 44, 47, 48, 50, 64,
 67, 68, 70, 72, 73, 75, 78, 115, 116, 117,
 118, 119, 125, 141
 serial robots, vii, viii, 41, 42, 43, 62, 103,
 104, 111
 service robots, 101, 102, 104, 141
 shape, 109, 112, 115, 131, 145, 147

signals, 3, 16, 116
 simulations, ix, 3, 6, 44, 67, 77, 144, 145,
 158, 160, 162, 163, 164
 society, 135
 solution, viii, 2, 11, 13, 18, 20, 24, 27, 38,
 42, 50, 77, 116, 117, 119, 154, 155, 156
 specifications, 42
 spindle, 113, 115, 116
 stability, 2, 5, 6, 35, 36, 102, 105, 106, 107,
 108, 111, 119, 124, 125, 126, 129, 131,
 132, 133, 134, 138
 state(s), 3, 4, 7, 8, 15, 16, 17, 18, 19, 20, 21,
 22, 23, 24, 25, 26, 27, 109, 114, 129, 157
 statically stable walking, 101, 111, 132,
 133, 137, 138
 structure, vii, viii, ix, 2, 4, 9, 14, 17, 27, 41,
 42, 44, 48, 81, 82, 83, 84, 86, 89, 96, 97,
 101, 102, 103, 106, 108, 110, 112, 113,
 114, 139
 structure synthesis, 83, 85, 87, 89, 91, 93,
 94, 95, 97, 99
 synthesis, vii, viii, ix, 42, 81, 83, 87, 89, 96,
 98, 99

T

target, 27, 94, 116, 124, 125, 126, 127, 129,
 133, 134, 135, 136, 137, 138, 139, 142
 technical support, 136
 techniques, viii, 42, 43
 technology, 112
 testing, 144
 time periods, 156
 topology, 144
 tracking performance, vii, viii, 1, 3, 8, 25,
 29, 31, 33, 158
 tracks, 16, 24
 trajectory, ix, 5, 20, 29, 60, 62, 63, 64, 65,
 68, 70, 71, 72, 78, 124, 128, 129, 131,
 144, 145, 148, 149, 154, 155, 156, 159,
 160, 161, 162, 163
 transformation, 83, 99, 114, 152
 translation, 45, 57
 transmission, 83, 97, 110, 111
 transportation, 136, 141
 trigonometric functions, 49

V

validation, 60
variables, 43, 44, 45, 47, 48, 49, 50, 51, 53,
54, 55, 56, 57, 58, 59, 60, 61, 64, 68, 69,
87, 146
variations, 2, 4, 6, 67
vector, 9, 13, 15, 16, 17, 19, 20, 22, 43, 46,
49, 50, 51, 53, 55, 59, 71, 89, 91, 97, 119,
120, 122, 147, 150, 153, 156, 157, 158
velocity, ix, 8, 13, 21, 24, 27, 45, 46, 51, 52,
53, 54, 56, 57, 60, 64, 66, 71, 82, 83, 86,
89, 91, 94, 97, 102, 111, 115, 122, 124,
128, 132, 134

vibration, 103
vision, 104, 105, 112, 147

W

walking, vii, ix, 101, 102, 103, 105, 106,
111, 112, 116, 125, 127, 128, 129, 130,
131, 132, 133, 137, 138, 139, 140
walking scenarios, 101, 103, 128, 132, 138
weakness, 2
weight ratio, vii, viii, 42, 108
workforce, 105